

이동 컴퓨팅 환경에서 다양한 데이터 크기를 지원하는 브로드캐스트 스케줄링 기법

신 지현⁰, 이 종환, 이 중화, 이 철숙, 김 경석
부산대학교 전자계산학과
{jhshin, jhwlee, jhlee, cslee, gimgs}@asadal.pnu.edu

The broadcast Scheduling method supporting unequal size of data in mobile computing environment

Shin Jihyeon, Lee Jonghwan, Lee Junghwa, Lee Cheolsuk, Kim Kyongsog
Department of Computer Science, Pusan National University

요약

이동 컴퓨팅은 주로 서버에서 클라이언트로의 대역폭이 상대적으로 큰 비대칭적 환경에서 이루어진다. 비대칭적 환경에서 정보 전달은 브로드캐스팅이 효과적이다. 기존의 많은 논문들이 서버가 전달하는 데이터의 크기가 같다고 가정하고 스케줄링 알고리즘을 제안하였지만 실제 데이터의 크기는 매우 다양하다. 인기도가 비슷하더라도 데이터 크기가 다를 때 큰 데이터 대신 작은 데이터 여러 개를 전송하면 클라이언트의 평균 대기시간이 감소하여 성능이 향상될 수 있다. 따라서 본 논문에서는 대기시간을 최소화하기 위해 인기도뿐만 아니라 데이터 크기도 함께 고려하는 스케줄링 기법을 제안하고 성능을 평가하였다.

1. 서론

이동 컴퓨팅 환경은 대역폭이 한정되고 전송 에러율이 높을 뿐 아니라 이동 컴퓨터의 전력이 제한되어 있는 등 많은 제약이 있다. 이동 컴퓨팅 환경은 주로 클라이언트에서 서버 방향(upstream)의 대역폭보다 서버에서 클라이언트 방향(downstream)의 대역폭이 상대적으로 훨씬 큰 비대칭적 통신 환경이다[1]. 이러한 통신환경에서 정보전달은 브로드캐스팅을 사용하는 것이 효과적이다. 브로드캐스팅은 클라이언트의 수에 상관없이 서비스를 확장할 수 있고, 한번의 전송으로 많은 요구를 만족시킨다는 장점이 있다. 브로드캐스팅은 유선 환경에서도 이미 여러 응용에서 이용되고 있다. 이동 컴퓨팅 환경에서는 전송할 데이터를 효과적으로 선택하여, 이동 클라이언트의 전력 소모량과 대기시간을 최소화하는 이동 컴퓨팅 환경에 적합한 브로드캐스팅 방법을 설계해야 한다[5].

브로드캐스팅을 연구한 많은 논문이 서버가 제공하는 데이터의 크기는 모두 같다고 가정하였다[1, 2, 3, 4, 5, 6]. 그러나 서버가 클라이언트에 전달하는 데이터의 크기는 다양하다. 따라서 데이터의 크기가 다를 때 한 주기에 할당되는 데이터 개수가 달라질 수 있고 방송주기의 길이 자체도 달라질 수 있다. 또한, 인기도가 비슷하더라도 데이터 크기가 다를 경우 데이터가 주기 중 어디에 위치하는가에 따라 대기시간이 달라진다. 그러므로 모든 데이터의 크기가 같다고 가정한 기존 연구의 스케줄링 기법은 크기가 다양한 경우에 적합하지 않다. 따라서 서버가 방송할 데이터를 선택하고 선택한 데이터를 브로드캐스트 채널에 할당하는 일은 클라이언트의 요구와 데이터의 인기도, 데이터의 길이를 고려하여 동적으로 이루어져야 한다.

본 논문에서는 브로드캐스트할 데이터를 선택할 때 인기도뿐만 아니라 데이터의 크기도 함께 고려하는 브로드캐스팅 구조와 서버의 스케줄링 기법을 제안한다. 그리고 데이터 크기가 다양한 환경에서 데이터의 인기도만을 고려하는 기존의 방법과

제안하는 기법의 클라이언트 대기시간과 전력 소모량을 측정하여 성능을 비교하였다.

2. 관련연구

최근 이동 컴퓨팅 분야에서 데이터 브로드캐스팅 방법에 대한 많은 연구가 이루어지고 있다. 연구분야는 크게 데이터 전달을 위한 브로드캐스팅 방식과 스케줄링 알고리즘으로 나눌 수 있다. 브로드캐스팅 방식 중 하나는 주기적 브로드캐스팅으로, 서버가 전송할 데이터를 선택한 후 주기적으로 데이터를 전송하는 기법이다[1, 2]. 이 방법은 인기가 높은(hot) 데이터를 자주 전송하여 인기있는 데이터를 원하는 클라이언트에게 유리하다. 그러나 인기가 낮은(cold) 데이터를 원하는 클라이언트는 데이터가 전송 될 때까지 채널을 관찰해야 한다. 또한 클라이언트가 필요로 하지 않는 데이터를 브로드캐스트하여 대역폭을 낭비할 가능성이 있다. 이것을 해결하기 위해 주문형 브로드캐스팅 기법이 연구되었다[3, 4]. 이 방법은 사용자의 요구를 모아서 데이터를 브로드캐스트 주기에 할당하여 한번의 전송으로 여러 클라이언트의 요구를 만족시킨다. 클라이언트가 요구하지 않은 데이터는 전송하지 않으므로 대역폭을 절약할 수 있는 장점이 있다. 그러나 주문형 기법은 짧은 시간에 많은 클라이언트의 요구가 집중되면 일부 클라이언트의 요구가 서비스되지 않을 수 있다. 또한 클라이언트가 브로드캐스트 채널에서 능동적으로 데이터를 찾는 것보다 서버에 요구를 전송하는 것이 더 많은 전력을 소비한다[5]. 클라이언트의 전력 소비량을 최소화하기 위해서는 요구 회수를 최소화해야 한다. 따라서 위의 두가지 방식을 혼합한 하이브리드(hybrid) 방식에 대한 연구도 많이 진행되고 있다[5, 6].

Acharya는 [7]에서 주문형 브로드캐스팅에서 데이터 크기에 따른 서비스 시간을 고려하여 EDF를 기반으로 한 스케줄링 기법을 제안하였다.

본 논문은 데이터 크기와 인기도를 함께 고려하는 스케줄링 기법과, 하이브리드 브로드캐스팅 방식을 이용하여 클라이언트의 대기시간과 전력 소모량을 줄이는 방안을 제안한다.

3. 다양한 크기의 데이터를 지원하는 브로드캐스팅 기법

3.1 브로드캐스팅 구조

본 논문에서 제안하는 브로드캐스트 채널 구조는 그림 1과 같다. 인덱스는 데이터 아이디와 데이터가 전송될 시간으로 구성되며, 각 주기동안 인덱스 채널을 통해 계속 브로드캐스트된다. 인덱스 채널을 따로 유지하면 인덱스가 전송될 때까지 기다리는 시간을 절약할 수 있다. 클라이언트는 필요한 데이터를 받기 위해 먼저 인덱스에서 검색하여 데이터가 전송되는 시간을 알아낸다. 만일 인덱스에서 데이터를 발견하면 전송이 시작되는 시간까지 대기모드(idle mode)로 전환하여 기다린 후 전송이 시작될 때 활동모드(active mode)로 전환하여 데이터를 받아온다. 만일 인덱스에 원하는 데이터가 없으면 서버에 요구(request mode)한 후 다음 주기부터 인덱스를 관찰한다.

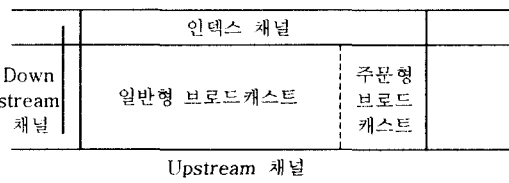


그림 1. 브로드캐스팅 구조

주문형 브로드캐스트 데이터는 일반형 브로드캐스트 데이터 다음에 할당된다. 주문형 브로드캐스트를 위한 논리적인 채널을 나눌 수 있지만 사용자의 요구가 적으면 채널 용량이 남아서 대역폭을 낭비하는 단점이 있다.

3.2 데이터 크기를 고려하는 스케줄링 알고리즘

서버가 클라이언트로 전달하는 데이터의 종류는 다양하고, 크기도 다르다. 서버가 전달하는 정보는 신문기사, 날씨, 증권, 교통 정보 등이 있다. 신문기사의 경우 이미지를 포함하고 있는가의 여부에 따라 크기가 달라진다. 이미지가 없는 기사는 1~2KB 정도이고 이미지가 있는 기사는 30~50KB 정도이다.

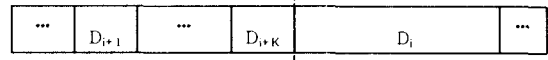
예를 들어 D_i 는 이미지를 포함하는 데이터이고 $D_{i+1} \sim D_{i+k}$ 은 텍스트로 구성된 데이터일 때, 각 데이터를 원하는 클라이언트의 수를 $P_{i+1} \sim P_{i+k}$ 라고 하자. P 는 인기를 나타내며, P 만 고려 즉, 인기도만 고려하여 우선순위를 결정했을 때 순위는 그림 2의 (a)와 같다고 가정한다. 이 경우 D_i 와 $D_{i+1} \sim D_{i+k}$ 의 인기도는 큰 차이가 나지 않을 것이다. 이처럼 큰 데이터와 작은 데이터의 인기도가 비슷할 때 (b)와 같이 D_i 대신 $D_{i+1} \sim D_{i+k}$ 을 브로드캐스트하면 만족시킬 수 있는 클라이언트의 수가 증가한다.

$$\sum_{i+1}^k P_{i+k} > P_i$$

브로드캐스트 한 주기로 충족시킬 수 있는 클라이언트가 많아지면 결과적으로 평균 대기시간이 감소하게 된다. 그러므로 인기도 차이가 작을 때 큰 데이터 대신 작은 데이터 여러개를 전송하는 것이 더 효율적이다.



(a) 인기도만 고려한 우선순위



브로드캐스트 / 브로드캐스트하지 않음

(b) 인기도와 크기를 함께 고려한 우선순위

그림 2. 브로드캐스트 스케줄을 결정하기 위한 우선순위

위의 예처럼 데이터의 크기가 다를 때 인기도만 고려하여 스케줄을 결정하면 평균 대기시간이 더 길어질 수 있다. 따라서 스케줄을 결정할 때는 데이터의 인기도뿐만 아니라 데이터의 크기도 함께 고려해야 한다.

그림 3은 데이터의 우선순위를 결정하기 위해 고려해야 할 데이터의 상대적인 크기와 데이터의 인기도를 반영하기 위한 온도 요소(TF: Temperature Factor)를 구하는 식이다. 인기도는 클라이언트의 실제 요구 회수를 고려해야 하지만 브로드캐스트 되는 동안 요구가 발생하지 않으므로 온도를 이용한다.

데이터 아이템 i ($i = 1 \dots N$)의 크기를 $S(i)$ 이라 할 때

$$SR(i) = \frac{S(i)}{\max(S(i))}$$

$$TF_n(i) = TF_{n-1}(i) + R_{n-1}(i) - \left[\frac{T_{cur} - TS_{n-1}(i)}{\lambda} \right]$$

$$TS_n(i) = \left[\frac{T_{cur} - TS_{n-1}(i)}{\lambda} \right] \cdot \lambda - TS_{n-1}(i)$$

- $SR(i)$: 데이터의 상대적인 크기
- $TF_n(i)$: 브로드캐스트 주기 n 에서 데이터 i 의 온도
- $R_{n-1}(i)$: 이전 주기에 발생한 클라이언트의 요구 회수
- λ : 클라이언트가 한 생애 머무는 평균 시간
- $TS_{n-1}(i)$: λ 시간 단위에 대한 마지막 timestamp
- $TS_n(i)$: 새로운 timestamp

그림 3. 데이터의 상대적 크기와 온도 요소 결정식

브로드캐스트 채널에 데이터를 할당하는 순서를 결정하기 위해 우선순위(PF: Priority Factor)를 구한다. 우선순위는 그림 4와 같이 일반(General) 브로드캐스팅을 위한 것과 주문형(On-demand) 브로드캐스팅을 위한 것으로 나누어 유지한다.

일반형(General) : $GPF_n(i) = \frac{TF_n(i) + W_T}{SR(i) + W_S}$

주문형(On-demand) : $AccumRequest(i) = 0$ 이면 $OPF_n(i) = 0$

$$OPF_n(i) = \frac{AccumRequest(i) + W_T}{SR(i) + W_S}$$

- $W_T + W_S = 일정$: 조정값
- $AccumRequest(i)$: 데이터 i 가 마지막으로 브로드캐스트된 이후에 발생한 요구 회수 (브로드캐스트되면 0으로 초기화)

그림 4. 우선순위 요소 (Priority Factor)

브로드캐스트 채널은 그림 1과 같이 일반형 영역과 주문형 영역으로 구성된다. 주문형 영역의 크기는 $OPF_n(i)$ 값이 큰 데이터부터 데이터의 크기를 더할 때, 최대 주문형 브로드캐스트

크기를 넘지 않을 때까지의 합이다. 일반형 영역의 크기는 브로드캐스트 주기의 크기에서 주문형 브로드캐스트 크기를 뺀 나머지가이다. 이와 같이 영역의 크기를 결정한 후 각 영역에 $GPF_n(i)$, $OPF_n(i)$ 값이 큰 데이터부터 차례로 배치한다.

4. 성능평가

제안한 스케줄링 기법과 온도만 고려하는 스케줄링의 성능을 비교하기 위하여 데이터 크기를 다양하게 분포시키면서 시뮬레이션하였다. 처음 20 주기는 서버의 $TF(i)$ 값을 초기화하기 위한 것으로, 이 주기의 대기시간과 전력 소비량은 무시하였다. 그리고 클라이언트는 zipf 분포에 따라 데이터에 접근한다고 가정하였다. 표 1은 실험에서 사용한 변수이다.

표 1. 시뮬레이션 변수

변수		값
서버	데이터 개수	1000
	데이터 크기	1KB~50KB
	브로드캐스트 주기	전체 데이터 크기의 10%
	시뮬레이션 시간	400 주기
클라이언트	λ	4000s~8000s
	Zipf skew	0.1~1.0
	Request interval	0.01s~1.0s
	idle, active, request 각 모드의 전력 소모량	0.1, 0.2, 0.5 (watt/s) (Motorola wireless PCMCIA modem, 1997)
전송채널	브로드캐스트 채널 대역폭	32 Kbps
	최대 주문형 브로드캐스트 비율	주기의 15%

그림 5와 그림 6은 zipf skew를 증가시키면서 클라이언트의 평균 대기시간과 평균 전력 소비량을 측정된 결과이다.

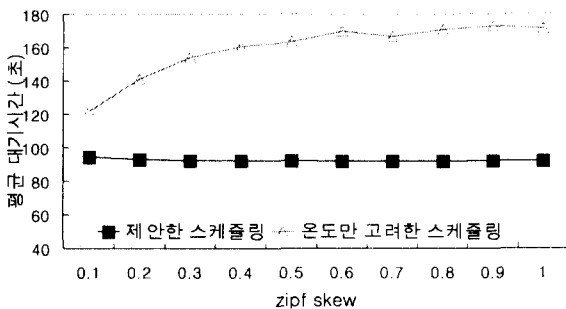


그림 5. 평균 대기시간

실험에서 한 주기에 할당되는 데이터 수를 측정된 결과 제안 기법의 경우 약 130개(평균크기보다 작은 것은 약 110개)였고, 온도만 고려한 경우는 약 60개(평균크기보다 작은 것은 약 33개)였다. 따라서 본 논문에서 제안한 기법은 온도가 비슷할 때 크기가 큰 데이터 하나를 전송하는 대신 작은 데이터 여러개를 전송하여 한 주기에 할당되는 데이터의 개수가 많아져서 더 많은 사용자 요구를 충족시킬 수 있었다. 따라서 그림 5와 같이 zipf skew가 커지면서 평균 대기시간이 조금씩 감소했다. 그러나 온도만 고려한 기법은 zipf skew가 커질수록 평

균 대기시간이 점차 증가하였다. 또한 온도만 고려한 경우 평균 대기시간이 길어져서 전력 소비량도 크기를 고려한 경우보다 더 많았다.

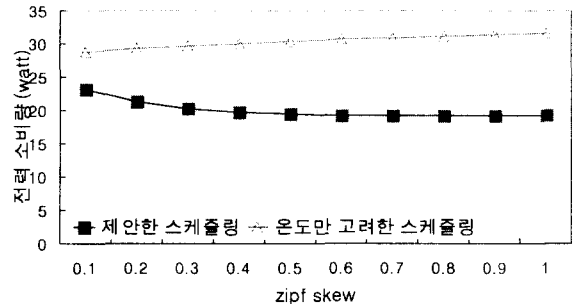


그림 6. 평균 에너지 소비량

5. 결론 및 향후 과제

기존의 많은 논문들이 서버가 전달하는 데이터의 크기가 같다고 가정하고 스케줄링 알고리즘을 제안하였지만, 실제로는 데이터의 크기는 다양하다. 성능평가에서 온도만 고려하는 방법보다 본 논문에서 제안한 데이터 크기를 함께 고려하는 스케줄링 기법이 더 효과적임을 보였다.

향후 연구에서는 큰 데이터를 작은 조각으로 잘라 여러번에 걸쳐서 전송하는 기법과 이 데이터 조각을 관리하기 위한 효율적인 인덱싱 기법을 연구할 예정이다. 또한 클라이언트의 효과적인 데이터 검색을 위해 데이터의 아이디뿐만 아니라 키워드로 인덱스를 구성하는 방법을 연구할 것이다.

참고문헌

- [1] Acharya, S. "Broadcast Disks: Data Management for Asymmetric Communications Environments", SIGMOD Conference, 199-210, 1995
- [2] Su, C.J. Tassiulas L. "Designing Broadcast Schedules for Information Dissemination through Broadcasting", TR, <http://www.isr.umd.edu/CSHCN/>, 1997
- [3] Aksoy, D. Franklin M. "Scheduling for Large-Scale On-Demand Data Broadcasting" INFOCOM '98 vol.2 651-659 1998
- [4] Xuan, P. "Broadcast on demand: efficient and timely dissemination of data in mobile environments", Real-Time Technology and Applications Symposium, 38-48, 1997
- [5] Datta, A. "Broadcast Protocols to Support Efficient Retrieval from Databases by Mobile Users", ACM Transactions on Database Systems, Vol. 24, No. 1, 1-79, 1999
- [6] Hu, J.H. "A Novel Push-and-Pull Hybrid Data Broadcast Scheme for Wireless Information Networks" ICC 2000, vol.3, 1778 -1782, 2000
- [7] Acharya, S. "Scheduling On-demand Broadcast: New Metrics and Algorithms", MOBICOM, 43-54, 1999