

이동 컴퓨팅 환경에서 데이터 크기를 고려하는 적응적 브로드캐스팅 기법

(An Adaptive Method based on Data Size for Broadcast in Mobile Computing Environments)

유영호^{*} 이종환^{**} 김경석^{***}

(Young-Ho Yu) (Jong-Hwan Lee) (Kyong-Sok Kim)

요약 : 이동 컴퓨팅은 최근 이동 통신 장비의 발전과 이동 통신 환경의 인터넷 연동으로 인해 컴퓨팅 연구 분야의 새로운 이슈로 등장했다. 이동 통신 환경은 잦은 접속 단절, 한정된 대역폭, 이동 클라이언트의 에너지 용량 제한 등 많은 제약이 있다. 이러한 제약으로 인해 많은 응용에서 사용자에게 데이터를 효율적으로 전달하기 위해 일반적으로 브로드캐스팅을 사용한다. 본 논문에서는 인기도와 데이터 크기를 고려하여 브로드캐스트하는 주기적 브로드캐스팅 채널과 이동 클라이언트의 요구에 따라 브로드캐스트하는 주문형 브로드캐스팅 채널로 브로드캐스팅 채널을 논리적으로 분할하고, 각 채널의 대역폭을 동적으로 할당하는 적응적 브로드캐스팅 기법을 제안한다. 제안하는 브로드캐스팅 기법은 브로드캐스트될 데이터를 선택할 때, 이동 클라이언트의 이동성을 고려하며, 본 논문에서 고안된 크기 요소 SF(Size Factor)를 사용하여 대상 데이터의 크기를 고려한다. 본 논문에서는 또한 실험을 통해 이동 클라이언트의 전력 소모량을 측정하여 제안하는 브로드캐스팅 기법의 성능을 평가한다.

키워드 : 적응적 브로드캐스팅, 하이브리드 브로드캐스팅, 크기 요소, 이동 컴퓨팅

Abstract Mobile computing becomes a new issue of researches in computing due to the advances of mobile equipment and the connection with Internet. In mobile environment, there are many constraints such as limited bandwidth, intermittent disconnection, limited battery life, and so on. By these reasons, broadcasting has been generally used to disseminate data efficiently by the mobile applications. This paper proposes an adaptive broadcasting method which logically divides broadcast channel into the periodic broadcast channel and the on-demand broadcast channel and dynamically assigns the bandwidths of both channel. The former disseminates data that are selected based on both the popularity and the size of each datum, the latter disseminates data that are selected based on the requests of mobile clients. When selecting data to be disseminated, the proposed broadcasting method considers the mobility of a mobile client and also considers the size of each datum by using SF(size factor) proposed in this paper. This paper also evaluates the proposed broadcasting method by measuring the energy expenditure of mobile client in experiments.

Key words : Adaptive Broadcasting, Hybrid Broadcasting, Size Factor, Mobile Computing

1. 서론

이동 컴퓨팅은 최근 이동 통신 장비의 발전과 이동 통신 환경의 인터넷 연동으로 인해 최근 가장 빨리 발전하고 있는 컴퓨팅 분야 중 하나이다. 이동 컴퓨팅 환경은 일반적으로 클라이언트에서 서버 방향(upstream)의 대역폭보다 서버에서 클라이언트 방향(downstream)의 대역폭이 상대적으로 훨씬 큰 비대칭적 통신 환경이다[1]. 비대칭적 통신 환경에서는 일정수의 서버가 많은 클라이언트에게 정보를 전달해야 하므로 브로드캐스팅

* 정회원 : 부산대학교 전자계산학과
yhyou@asadal.pnu.edu

** 비회원 : 부산대학교 전자계산학과
jhwlee@asadal.pnu.edu

*** 종신회원 : 부산대학교 전자전기정보컴퓨터공학부 교수
gimsg@asadal.pnu.edu

논문접수 : 2002년 2월 8일

심사완료 : 2003년 1월 7일

기법을 사용하는 것이 효과적이다. 브로드캐스팅은 클라이언트의 수에 상관없이 서비스를 확장할 수 있고, 한번의 전송으로 많은 요구를 만족시킨다는 장점이 있다. 이동 컴퓨팅 환경에서는 브로드캐스트할 데이터를 효과적으로 선택하여, 이동 클라이언트의 전력 소모량과 데이터 액세스 시간(access time)을 최소화하는 브로드캐스팅 기법을 설계해야 한다.

이동 통신 환경과 인터넷 연동으로 인하여 이동 컴퓨팅의 응용은 다양한 영역으로 확대되고 있다. 이에 따라 응용에서 다루는 데이터의 수와 종류도 다양해지고 있다.

브로드캐스트될 데이터의 수가 많을 경우, 각 데이터의 인기도에 따라 데이터를 브로드캐스트함으로써, 브로드캐스팅 기법의 성능을 향상시킬 수 있다. 그러나, 인기도가 낮은 데이터에 대해서는 소외 현상(starvation)이 발생할 수 있는 단점이 있다. 이에 대한 방안으로 하이브리드 브로드캐스팅 기법이 제안되었다. 하이브리드 브로드캐스팅 기법은 인기도에 따라서 브로드캐스트할 데이터를 선택하고, 브로드캐스트 채널의 일정 부분을 클라이언트의 요구에 따라 브로드캐스트하도록 한다.

데이터의 종류가 다양해지면 데이터의 크기도 다양해진다. 데이터의 크기는 브로드캐스팅 기법의 성능에 중요한 영향을 준다. 인기도가 비슷하고 데이터의 크기가 다를 경우에 큰 데이터 하나 대신 작은 데이터 여러 개를 브로드캐스트하는 것이 더 효율적일 수 있다. 또한, 이동 컴퓨팅 환경에서 이동 클라이언트는 자유롭게 이동하므로 이동 클라이언트가 서버에 요청한 데이터는 이동 클라이언트가 서버가 담당하는 셀 영역을 빠져나가기 전에 서비스되어야 한다.

따라서, 본 논문에서는 브로드캐스트될 데이터의 크기와 이동 클라이언트의 이동성을 고려하는 하이브리드 방식의 적응적 브로드캐스팅 기법을 제안한다. 제안하는 브로드캐스팅 기법은 브로드캐스팅 채널을 주기적 브로드캐스팅 채널과 주문형 브로드캐스팅 채널로 논리적으로 분할하여 사용한다. 분할된 채널의 대역폭은 클라이언트가 요구하는 데이터의 수와 종류에 따라 동적으로 할당된다. 또한 각 채널에서 브로드캐스트될 데이터를 선택할 때, 본 논문에서 고안된 크기 요소를 사용하여 데이터의 크기에 따라 데이터의 우선순위를 차별적으로 고려할 수 있도록 한다. 주문형 채널에서 브로드캐스트되는 데이터를 선택할 때는 이동 클라이언트의 이동성을 고려하여 이동 클라이언트가 셀 안에 머무르는 시간에 방송될 수 있는 경우에만 선택되도록 한다.

또한, 본 논문에서는 실험을 통해 이동 클라이언트의

전력 소비량을 측정하여 제안하는 브로드캐스팅 기법의 성능을 평가한다. 실험 결과를 통해 데이터의 크기를 고려하는 것이 브로드캐스팅 기법의 성능에 효과가 있음을 보이고, 또한, 주기적 브로드캐스팅 채널과 주문형 브로드캐스팅 채널의 대역폭을 동적으로 할당하는 것이 효과가 있음을 보인다.

2. 관련연구

최근 이동 컴퓨팅 분야에서 데이터 전달(dissemination) 방법에 대한 많은 연구가 이루어지고 있다. 이동 컴퓨팅 환경은 일반적으로 비대칭적 통신 환경이다[1]. 비대칭적 통신 환경에서는 일정수의 서버가 많은 클라이언트에게 정보를 전달해야 하므로 데이터 전달 방법으로 브로드캐스팅 기법을 많이 사용한다. 브로드캐스팅 기법에 대한 연구들은 크게 브로드캐스트 모델, 스케줄링 기법, 인덱싱 기법으로 나눌 수 있다.

2.1 브로드캐스 모델과 스케줄링 기법

현재까지 연구되고 있는 데이터 전달 기법으로는 주문형 클라이언트/서버, 주기적 브로드캐스팅, 주문형 브로드캐스팅, 하이브리드 브로드캐스팅 기법 등이 있다. 이 기법들은 그림 1과 같이 전송 방식과 클라이언트와의 상호작용 유무에 따라 데이터 전달 방법이 달라진다.

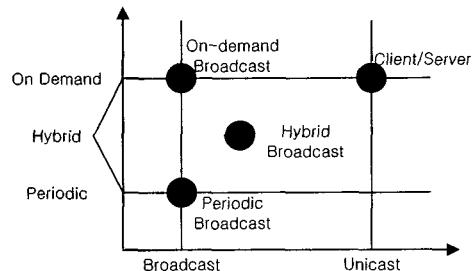


그림 1 데이터 전달 기법

클라이언트/서버 모델에서 클라이언트는 서버에 서비스를 요청(on-demand)하고 서버는 한 시점에 단일 클라이언트에만(unicast) 서비스를 제공한다. 클라이언트/서버 모델은 클라이언트의 주문에 따라 서비스 할 수 있는 장점이 있다. 그러나 짧은 시간에 많은 클라이언트의 요구가 집중되면 성능이 심각하게 저하될 수 있다.

2.1.1 주기적 브로드캐스팅(Periodic Broadcasting)

주기적 브로드캐스팅 모델에서 서버는 한 시점에 여러 클라이언트에 서비스를 제공한다. 이 방법은 많은 정

보를 액세스할 때 클라이언트의 전력 소모가 적으며, 인기가 높은(hot) 데이터를 자주 전송하여 인기 있는 데이터를 원하는 클라이언트에게 유리하다. 그러나 인기가 낮은(cold) 데이터를 원하는 클라이언트는 데이터가 전송 될 때까지 채널을 관찰해야 한다. 또한 데이터베이스에 있는 모든 데이터를 전송하므로 클라이언트가 필요로 하지 않는 데이터도 브로드캐스트하여 대역폭을 낭비할 가능성이 있다.

주기적 브로드캐스팅의 대표적인 기법이 브로드캐스트 디스크(broadcast disk)이다[1]. 이 기법은 데이터베이스에 있는 모든 데이터를 포함하는 브로드캐스트 순서를 결정하여 주기적으로 전송하는 기법으로 클라이언트가 하나의 저장 장치처럼 사용하도록 한다. 또한 메모리 계층구조 개념을 이용하여 데이터의 중요성에 따라 대역폭을 다르게 할당하는 멀티 레벨(multi-level) 방식이다. 그러나 브로드캐스트 디스크는 이미 알려진 액세스 확률에 따라 구성된다. 그리고 브로드캐스트 순서가 결정된 후에는 브로드캐스트 내용이 바뀌지 않을 뿐만 아니라 클라이언트와 상호작용이 없으므로 클라이언트 액세스 패턴의 변경 가능성을 고려하지 않았으며 이동 클라이언트의 이동성(mobility)도 고려하지 않았다.

2.1.2 주문형 브로드캐스팅(On-Demand Broadcasting)

주문형 브로드캐스팅은 서버가 특정 클라이언트의 요구를 받아서 여러 클라이언트에게 서비스하는 기법이다. 이 방법은 사용자의 요구를 모아서 한 번의 전송으로(batch) 여러 클라이언트의 요구를 만족시킬 수 있다. 따라서 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.

Xuan은 클라이언트의 요구에 따라 브로드캐스트하는 모델인 BoD(Broadcast on Demand)를 제안하고 클라이언트/서버 모델, 주기적 브로드캐스팅 기법 등과 성능을 비교하였다[2]. 그리고 BoD를 위한 스케줄링 정책들을 비교, 평가하였다. Aksoy는 주문형 브로드캐스팅 기법에서 대역폭, 클라이언트의 수, 데이터 개수에 확장성이 있고 개별 클라이언트와 전체 성능간의 균형을 유지할 수 있는 온라인 스케줄링 알고리즘을 제안하였다[3]. 그러나 이들 연구는 클라이언트의 액세스 시간을 통해 성능을 평가하였고 이동 클라이언트가 소비하는 전력에 대해서는 고려하지 않았다. 클라이언트가 브로드캐스팅 채널에서 능동적으로 데이터를 찾는 것보다 서버에 요구를 전송하는 것이 더 많은 전력을 소비한다[4]. 클라-

이언트의 전력 소비량을 최소화하기 위해서는 요구회수를 최소화하도록 방송 순서를 작성해야 한다.

2.1.3 하이브리드 브로드캐스팅(Hybrid Broadcasting)

하이브리드 브로드캐스팅 기법은 주기적 브로드캐스팅 방식과 주문형 브로드캐스팅 방식을 함께 이용하는 것이다. 자주 액세스될 것으로 예상되는 데이터는 주기적으로 서비스하고 주기적 브로드캐스팅을 통해서 만족되지 못한 클라이언트의 요구는 주문형 브로드캐스팅 방식을 통해 서비스된다.

일반적으로 주기적 브로드캐스팅 기법은 데이터베이스에 있는 모든 데이터를 전송하지만 데이터 개수가 많아지면 주기가 너무 길어져 액세스 시간이 증가하므로 실제로 모든 데이터를 전송할 수 없다. 따라서 서버는 적절한 스케줄링 알고리즘을 통해 전송할 데이터를 선택해야 한다.

지금까지 평균 액세스 시간을 최소화하는 많은 기법들이 제안되었다[5,6,7]. 그러한 연구 중 대부분이 데이터의 인기도(popularity)나 데이터의 액세스 빈도(access frequencies)를 기초로 하여 전송할 데이터를 선택하였다. 일부 연구는 실시간 데이터를 전송하기 위해 EDF(Early Deadline First)를 기반으로 한 데이터 선택 기법에 대해 연구하였다[8]. 그러나 이 연구들은 데이터의 크기가 모두 같다고 가정하였다. 그러나 실제로 서버가 클라이언트에 전달하는 데이터의 크기는 다양하다. 데이터의 크기가 다를 때 한 주기에 할당되는 데이터 개수가 달라질 수 있고 브로드캐스트 주기의 길이 자체도 달라질 수 있다. 또한 인기도가 비슷하더라도 데이터 크기가 다를 경우에 큰 데이터 하나 대신 작은 데이터 여러 개를 전송하는 것이 더 효율적일 수도 있다. 그러므로 모든 데이터의 크기가 같다고 가정한 기존 연구의 데이터 선택 기법은 크기가 다양한 경우에 적합하지 않다. 따라서 본 논문에서는 서버가 브로드캐스트 할 데이터를 선택할 때, 인기도(온도)뿐만 아니라 데이터 크기를 함께 고려한다.

2.2 인덱싱 기법

브로드캐스팅을 위한 인덱싱 기법은 클라이언트의 액세스 시간을 짧게 유지하는 동시에 전력 소비량을 줄이기 위해 소개되었다. 브로드캐스팅을 위한 인덱싱 기법들([9,10])은 기본적으로 전송할 데이터의 도착 시간에 대한 정보를 포함하여 이동 클라이언트가 원하는 데이터의 도착 시간을 예측할 수 있도록 한다. 본 논문에서는 기존의 (1:m) 인덱싱 기법을 이용하였다.

[10]의 (1:m) 인덱싱 기법을 그림 2에 나타내었다. 인덱스 세그먼트(index segment)는 연속적인 인덱스 베kt

(index buckets)의 집합이고 데이터 세그먼트(data segment)는 인덱스 세그먼트 사이에서 브로드캐스트되는 데이터 버킷(data buckets)의 집합이다. (1:m) 인덱싱은 브로드캐스팅 한 주기동안 인덱스를 m번 전송하는 방식이다. 데이터는 키 순서로 전송되며, 데이터 중 $1/m$ 을 전송할 때마다 인덱스는 전송된다. 각 인덱스 버킷은 <키, 포인터> 리스트로 구성되며 다음 인덱스 세그먼트의 시작을 가리키는 오프셋(offset)을 포함한다. 그리고 각 데이터 버킷은 다음 인덱스 세그먼트의 시작을 가리키는 오프셋을 포함한다.



그림 2 (1:m) 인덱싱

(1:m) 인덱싱 기법에서는 데이터는 키 순서로 정렬되어 전송된다. 그러나 평균 액세스 시간을 줄이기 위해서 액세스 빈도가 높은 데이터를 브로드캐스팅 주기의 앞쪽에 배치해야 한다. 본 논문에서는 데이터는 주기적 브로드캐스팅 데이터를 우선순위 값 순서로 먼저 배치한 다음에 주문형 브로드캐스팅 데이터를 우선순위 값 순서로 배치한다.

기존의 (1:m) 기법은 인덱스 버킷의 포인터는 데이터의 도착시간이나 다음 인덱스를 가리킨다. 또한 데이터 베이스에 있는 모든 데이터가 전송된다고 가정하였다. 그러나 본 논문에서는 모든 데이터를 전송하지 않으므로 데이터가 브로드캐스트되지 않는다면 포인터는 널(null)로 설정하여 클라이언트의 액세스 시간을 줄일 수 있다.

3. 데이터 크기를 고려하는 적응적 브로드캐스팅 기법

브로드캐스트 스케줄링은 브로드캐스트 할 데이터를 선택하는 과정과 데이터를 채널에 배치하는 과정으로 나누어진다. 서론에서 살펴본 바와 같이 데이터의 크기가 다양한 환경에서 평균 액세스 시간을 최소화하기 위해서 브로드캐스트 할 데이터를 선택할 때는 데이터의 크기를 고려하고, 데이터를 채널에 배치하는 과정에서는 클라이언트의 액세스 시간이 짧도록 유지하면서 전력 소비량을 줄이기 위해 인덱스 트리를 전송한다. 또한, 주기적 브로드캐스팅 채널과 주문형 브로드캐스팅 채널

의 대역폭을 동적으로 할당하여 클라이언트의 요구에 적용할 수 있도록 한다.

브로드캐스팅 기법의 설계에서 스케줄링 기법과 함께 인덱싱 기법도 중요하다. 본 논문에서는 인덱싱 기법은 따로 고려하지 않고, [10]에서 제안된 (1:m) 인덱싱 기법을 약간 수정하여 사용한다.

3.1 이동 컴퓨팅 환경

일반적으로 이동 컴퓨팅 환경은 그림 3과 같이 무선 인터페이스를 가진 베이스 스테이션(base station) 서버(이하 서버)와 이동 클라이언트 등으로 구성된다. 그림에서 하나의 서버가 담당하는 영역을 셀(Cell)이라 한다. 이동 클라이언트는 자유롭게 셀에 들어오고 나간다. 따라서 이동 컴퓨팅 환경에서 브로드캐스팅 기법을 설계 할 때는 사용자의 이동성을 고려해야 한다. 서버는 실시간 데이터가 아니더라도 클라이언트가 데이터를 요청하면 셀을 벗어나기 전에 브로드캐스트하거나 이웃 셀로 클라이언트의 요구를 전달해야 한다. 본 논문에서는 셀을 벗어나기 전에 브로드캐스트 할 수 없는 데이터는 선택에서 제외하는 것으로 한다.

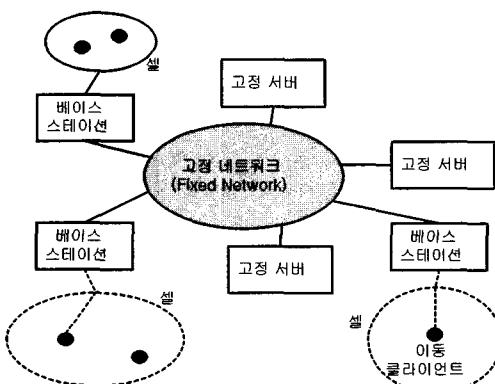


그림 3 이동 컴퓨팅 환경

본 논문에서 가정하는 브로드캐스팅 채널은 그림 4와 같이 구성한다.

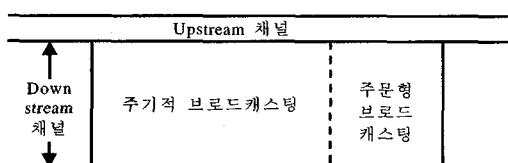


그림 4 브로드캐스팅 채널 구조

자주 액세스될 것으로 예상되는 데이터는 주기적 브로드캐스팅 채널을 통해 전송한다. 그리고 모든 데이터를 전송하지 않으므로 주기적 브로드캐스팅을 통해 전송되지 않는 데이터를 필요로 하는 클라이언트를 위해 Upstream 채널을 통해 요구를 받는다. 클라이언트가 요구한 데이터는 주문형 브로드캐스팅 채널을 통해 서비스된다. 주기적 브로드캐스팅 채널을 주기의 앞쪽에 배치하고 주문형 브로드캐스팅 채널은 그 다음에 배치한다. 두 채널의 대역폭은 서비스되지 않은 클라이언트의 요구 수와 데이터 수에 따라 동적으로 조절된다.

3.2 브로드캐스트할 데이터 선택 기법

기존의 연구에서 브로드캐스트할 데이터 선택은 주로 인기도에 기반한 우선순위에 따라 이루어졌다. 본 논문에서는 인기도뿐만 아니라, 데이터 크기에 따라 인기도를 차별적으로 적용하도록 하는 크기 요소를 고안하여 데이터 선택 기법에 사용한다. 데이터 선택은 주기적 브로드캐스팅을 위한 데이터 선택과 주문형 브로드캐스팅을 위한 데이터 선택으로 나누어진다. 먼저 데이터의 인기도를 예측하고, 이를 데이터의 크기 요소 값에 따라 차별적으로 적용하여 데이터를 선택한다. 주기적 브로드캐스팅을 위한 인기도는 클라이언트가 데이터를 요구할 확률을 이용하여 예측하고, 주문형 브로드캐스팅을 위한 인기도는 Upstream 채널을 통해 전달된 클라이언트의 요구 수를 적용한다. 이 값들과 크기 요소 값을 바탕으로 데이터의 각 채널에서의 우선순위를 결정한다. 데이터의 선택과 배치는 각 브로드캐스트 주기의 시작 전에 결정된다. 따라서 서버가 스케줄을 결정하는 동안 발생한 요구는 반영되지 않는다.

3.2.1 크기를 고려한 데이터 선택

서버가 클라이언트로 전송하는 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.

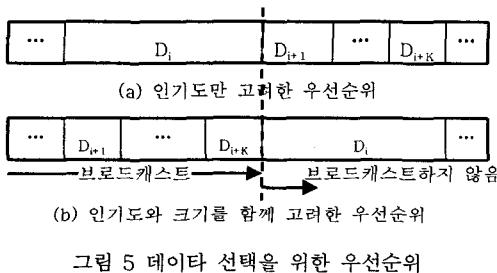


그림 5 데이터 선택을 위한 우선순위

예를 들어 D_i 는 이미지를 포함하는 데이터이고 $D_{i+1} \sim D_{i+K}$ 는 텍스트로 구성된 데이터일 때, 각 데이터를 원하는 클라이언트의 수를 $C_i \sim C_{i+K}$ 라고 하자. C_i 는 결국 인기도를 나타내며, C_i 만 고려할 경우, 즉, 인기도만 고려하여 우선순위를 결정했을 때 ($C_i \geq C_{i+1} \geq \dots \geq C_{i+K}$ 라고 가정) 우선순위는 그림 5의 (a)와 같다. 이 경우 D_i 와 $D_{i+1} \sim D_{i+K}$ 의 인기도, 즉 $C_i \sim C_{i+K}$ 는 큰 차이가 나지 않을 것이다. 이처럼 큰 데이터와 작은 데이터의 인기도가 비슷할 때 (b)와 같이 D_i 대신 $D_{i+1} \sim D_{i+K}$ 를 브로드캐스트하면 식 1과 같이 요구를 충족시킬 수 있는 클라이언트의 수가 더 많아질 수 있다.

$$\sum_{k=1}^K C_{i+k} > C_i \quad (1)$$

한 번의 브로드캐스트 주기에서 요구를 충족시킬 수 있는 클라이언트가 많아지면 결과적으로 평균 액세스 시간이 감소하게 된다. 그러므로 인기도 차이가 작을 때 큰 데이터 대신 작은 데이터 여러 개를 전송하는 것이 더 효율적일 수 있다.

위의 예처럼 데이터의 크기가 다를 때 인기도만 고려하여 스케줄을 결정하면 평균 액세스 시간이 더 길어질 수 있다. 따라서 본 논문에서는 데이터의 인기도뿐만 아니라 데이터의 크기도 함께 고려하여 브로드캐스트 스케줄을 결정하도록 한다.

본 논문에서는 데이터 크기를 반영하기 위해 각 데이터의 크기 요소(size factor) $SF(i)$ 를 고안하여 사용한다. 크기 요소는 브로드캐스트될 데이터를 선택할 때 사용하는 우선순위 값을 계산할 때, 온도 요소에 곱해져서 비슷한 인기도를 가지는 데이터들 중에서 크기가 작은 데이터가 선택되도록 한다. 만일 데이터 크기에 따라 크기 요소 값의 차이가 너무 크면 크기가 큰 데이터는 소외 현상이 발생할 수 있다. 소외 현상을 방지하기 위해 크기 요소 값은 큰 차이가 나지 않도록 해야 한다. 따라서 데이터 아이템 i 의 크기를 $S(i)$ 라고 할 때, 데이터의 크기 요소 $SF(i)$ 는 식 2와 같다. 여기서 c 는 정규화 상수로 데이터의 크기 차이가 너무 많이 날 경우, $SF(i)$ 의 값이 일정한 범위에 있도록 조정하는 역할을 한다. 상수 c 는 크기 요소 $SF(i)$ 가 너무 커지지 않도록 용용에서 사용하는 데이터의 크기 분포를 분석하여 적절히 조정하여 사용한다. 크기 요소가 너무 커지면 기존 우선 순위를 왜곡시킬 수 있다. 본 논문에서는 1로 하여 사용하였다. 식 2에서 \log 는 상용로그를 사용하며, 크기 요소값이 1보다 작아지지 않도록 하기 위해 크기 비율에 10을 더해서 로그를 취하도록 한다.

$$SF(i) = \log\left(\frac{\max\{S(1) \dots S(N)\}}{c \times S(i)} + 10\right) \quad (2)$$

3.2.2 우선순위 결정에 필요한 요소들

데이터의 우선순위를 결정하기 위해서는 데이터의 인기도, 크기 요소, 클라이언트가 셀에 머무는 시간 등이 요구된다. 본 논문에서는 데이터의 인기도를 반영하기 위해서 온도(temperature) 요소를 이용한다. 데이터의 인기도는 클라이언트의 액세스 패턴에 따라 변화하므로 온도 개념을 이용한다. 즉, 데이터를 요구하는 클라이언트가 많아지면 온도가 증가하고 브로드캐스팅을 통해 요구를 만족시킨 만큼 온도는 내려간다. 논문에서 사용하는 변수들을 표 1에 나타내었다.

표 1 데이터 선택에 사용하는 요소

요소	요소에 대한 설명
λ	클라이언트가 셀에 머무는 평균 시간
bcast	브로드캐스트 주기
P_i	데이터 i 의 인기도
$S(i)$	데이터 i 의 크기
$R(i)$	데이터 i 에 대한 요구 수
$AGE(i)$	데이터 i 가 연속해서 방송 안된 횟수
$IF(i)$	데이터 i 가 방송에서 제외된 정도 (ignore factor)
$TF(i)$	데이터 i 의 온도(temperature factor)
$OPF(i)$	데이터 i 의 주문형 채널에서 우선순위 (on-demand priority factor)
$PPF(i)$	데이터 i 의 주기적 채널에서 우선순위 (periodic priority factor)

λ 는 [4]에서 처음 이용한 개념이다. 서버는 데이터를 브로드캐스트하기 위해 선택할 때 데이터가 브로드캐스팅 데이터 집합에 남아있을 시간을 지정할 필요가 있다. 이 시간을 λ 라고 한다. 한 셀에 대해서 λ 는 이동 클라이언트가 그 셀에 머무는 평균 시간이라고 할 수 있다. 특정 셀에 대한 λ 값은 사용자의 이동 패턴과 셀의 위치, 그리고 응용의 특성에 대한 사전 지식에 기초하여 계산될 수 있다[4].

3.2.3 주기적 브로드캐스팅을 위한 데이터 선택

클라이언트의 액세스 시간은 원하는 데이터를 찾기 위해 브로드캐스팅 채널에 처음 접근할 때부터 데이터를 내려 받는 데까지의 시간을 의미한다. 주기적 브로드캐스팅 채널에서 클라이언트의 요구를 충분히 만족시킨

다면 많은 클라이언트들은 늦어도 두 주기 안에 원하는 데이터를 얻을 수 있다. 만일 원하는 데이터가 브로드캐스트되지 않으면 클라이언트는 서버에 요구 패킷을 보내고 다음 주기부터 데이터가 도착할 때까지 기다려야 하므로 액세스 시간이 길어진다. 그리고 요구 패킷을 보내는 요구모드는 활동모드나 휴면모드보다 더 많은 전력을 소비한다. 요구 패킷을 전송한 후에는 일반적인 데이터 탐색 과정을 다시 거쳐야 하므로 전력 소비량은 크게 증가한다. 이와 같이 주기적 브로드캐스팅을 위한 데이터 선택의 결과는 평균 액세스 시간과 전력 소비량을 좌우한다.

각 브로드캐스트 주기에서 데이터 온도 요소는 클라이언트가 셀에서 머무는 평균 시간인 λ 시간동안 데이터 D_i 를 원하는 클라이언트의 수를 이용한다. 클라이언트가 데이터 D_i 를 요구할 확률 P_i 를 인기도로 사용한다. λ 시간동안 데이터 D_i 에 대한 요구 수는 λ 시간동안 셀에 머무르는 클라이언트들의 요구 수의 합에 P_i 를 곱하여 얻을 수 있다. 요구 수를 그대로 온도로 이용하게 되면 항상 인기도가 높은 데이터만 선택이 되므로, 최초의 온도, 즉, 브로드캐스트되지 않는 데이터의 온도는 각 데이터 D_i 의 인기도 P_i 를 적용하여 구하고, 현재 브로드캐스트되고 있는 데이터의 온도는 원래온도에서 한 주기동안 만족시킨 요구 수만큼 빼도록 한다. λ 시간동안의 총 요구 수에 P_i 를 적용하여 구한 $TF_n(i)$ 는 λ 시간동안 데이터 D_i 에 대한 요구 수를 나타내게 된다. 한 주기 동안 만족시킬 수 있는 요구 수는 한 주기의 방송 시간을 bcast라고 할 때, λ 시간동안 요구 수의 bcast/ λ 이 된다. λ 시간동안 셀에 머무르는 평균 클라이언트의 수를 C_{avg}^λ 라고 하고, λ 시간동안 한 클라이언트의 평균 요구 수를 RQ_{avg}^λ 라고 하면, λ 시간동안의 총 요구 수는 $C_{avg}^\lambda \times RQ_{avg}^\lambda$ 로 구할 수 있다. 따라서, n번째 주기에서 각 데이터의 온도 요소 $TF_n(i)$ 는 식 3으로 구할 수 있다.

$$\text{if broadcasting, } TF_n(i) = TF_{n-1}(i) \times (1 - \frac{bcast}{\lambda}) \\ \text{else } TF_n(i) = C_{avg}^\lambda \times RQ_{avg}^\lambda \times P_i \quad (3)$$

온도 요소와 크기 요소 값을 결정한 후에는 데이터의 우선순위 값을 계산한다. n번째 주기에서 데이터 D_i 의 주기적 브로드캐스팅을 위한 우선순위 값은 식 4와 같다.

$$PPF_n(i) = \{TF_n(i) + AGE_n(i)\} \times SF(i) \quad (4)$$

$AGE_n(i)$ 는 데이터 D_i 가 연속적으로 브로드캐스트되지 않은 주기 수이다. 데이터 D_i 가 얼마나 오래 전송되

지 않았는지를 나타내는 값으로, 우선순위 값의 근소한 차이로 스케줄에 포함되지 못한 데이터를 간헐적으로 선택하여 소외 현상을 줄이기 위해 사용하는 변수이다. $AGE_n(i)$ 는 식 5로 구한다.

$$AGE_n(i) = n - LAST(i) \quad (5)$$

$LAST(i)$ 는 데이터 D_i 가 마지막으로 브로드캐스트된 주기 번호이다. 따라서 한번 브로드캐스트되면 $AGE_n(i)$ 의 값은 1로 감소한다. 각 데이터에 대한 우선순위 값 $PPF_n(i)$ 를 결정한 후에는 그 값에 따라 내림차순으로 정렬하여 우선순위 값이 큰 데이터부터 주기적 브로드캐스팅 채널 대역폭만큼 선택한다. 주기적 브로드캐스팅 채널의 대역폭은 전체 브로드캐스팅 채널의 대역폭에서 주문형 브로드캐스팅 채널의 크기를 뺀 나머지이다.

3.2.4 주문형 브로드캐스팅을 위한 데이터 선택

주문형 브로드캐스팅 채널은 클라이언트가 직접 요구한 데이터를 제공하기 위해 사용된다. 주기적 브로드캐스팅에서 발생할 수 있는 소외현상을 주문형 브로드캐스팅 채널에서 보완해야 한다. 따라서 주문형 브로드캐스팅 데이터는 EDF 알고리즘을 기반으로 선택해야 한다.

클라이언트는 여러 셀을 오고 가므로 하나의 셀에서 데이터를 요구한 후 셀을 벗어나기 전에, 즉, 평균 λ 시간 내에 요구된 데이터는 적어도 한번 서비스되어야 한다. 따라서 본 논문에서는 여러 주기동안 브로드캐스트되지 않은 데이터를 선택할 수 있도록 $IF(Ignore Factor)$ 를 적용하였다. $IF(i)$ 는 데이터 D_i 에 대한 요구가 도착한 이후 브로드캐스트되지 않은 주기 수로 정의한다.

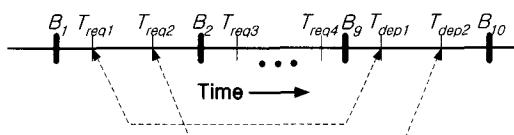


그림 6 Ignore Factor 계산

그림 6에서 $B_1, B_2, \dots, B_9, B_{10}$ 는 서버가 브로드캐스트 스케줄을 결정하는 지점이다. 즉 (B_1, B_2) 사이의 시간은 $bcast$ 이다. λ 가 $8 \times bcast$ 라고 가정할 때 B_{10} 지점에서 T_{req1} 과 T_{req2} 는 λ 시간이 지났으므로 무시된다. 즉 B_2 지점 이전에 도착한 요구는 모두 무시해야 한다. 현재 지점으로부터 $|\lambda/bacst|$ 이전 주기 이후에 도착한 요구 중 첫 번째 요구에 따라 $IF(i)$ 를 계산하면 된다.

따라서 서버는 현재 지점에서 $|\lambda/bacst|$ 앞 지점 이후 처음 도착한 요구의 도착시간만 유지하고 나머지는 무시해도 된다. 즉, $|\lambda/bacst|+1$ 개의 각 주기에서 각 D_i 에 대해 처음 도착한 요구의 시간정보만 유지하면 된다.

데이터 D_i 가 마지막으로 브로드캐스트된 시간이 T_0 이고 T_0 이후에 데이터 D_i 에 대한 첫 번째 요구가 도착한 시간을 T_{req} 라하고, 현재 시간 T_{cur} 시간까지 클라이언트의 아무런 요구가 도착하지 않았다고 가정할 때, T_{cur} 시간에 $IF(i)$ 는 식 6과 같다.

$$IF(i) = \left\lfloor \frac{T_{cur} - T_{req}}{bcast} \right\rfloor + 1 \quad (6)$$

그러나 이것은 $(T_{dep} - T_{req} \leq \lambda)$ 일 때만 유효하다. 예 들어 클라이언트 C 가 T_{req} 에 데이터를 요청한 후 T_{cur} 시간까지 다른 요구가 도착하지 않았다고 가정할 때, λ 가 클라이언트가 셀에 머무는 평균 시간이므로, 클라이언트 C 는 $T_{dep}(=T_{req} + \lambda)$ 이후에는 셀을 떠났을 걸로 예측할 수 있다. 이후에 다른 요구가 도착하지 않는다면 $IF(i)$ 는 1로 감소한다.

$$1 \leq IF(i) \leq \left\lfloor \frac{\lambda}{bcast} \right\rfloor + 1 \quad (7)$$

$IF(i)$ 를 결정한 후 서버는 주문형 브로드캐스팅을 위한 데이터를 선택하기 위해 우선순위를 결정한다. 우선순위 값은 식 8과 같이 각 D_i 에 대한 클라이언트의 요구 수 $R_{n-1}(i)$ 에 크기 요소와 IF 를 곱하여 구한다.

$$OPF_n(i) = IF(i) \times R_{n-1}(i) \times SF(i) \quad (8)$$

이 값에 따라 내림차순으로 정렬하여 우선순위 값이 큰 데이터부터 선택한다. 주문형 브로드캐스팅 채널의 대역폭은 최대 주문형 브로드캐스팅 채널의 대역폭 크기 제한을 넘지 않을 때까지 $OPF(i)$ 값에 따라 선택된 데이터 크기의 합이 된다.

3.3 동적인 주문형 브로드캐스팅 채널의 대역폭 결정

기존의 연구에서는 최대 주문형 브로드캐스팅 채널의 크기는 일정하게 유지하였다. 그런데 최대 주문형 브로드캐스팅 채널의 크기는 클라이언트의 액세스 시간에 영향을 준다. 주문형 브로드캐스팅 채널의 대역폭이 작으면 소외가 증가할 수 있고, 주문형 브로드캐스팅 채널의 대역폭이 너무 크면 주기적 브로드캐스팅 채널에서 만족하지 못한 클라이언트가 증가하여 클라이언트의 요구회수와 액세스 시간, 전력 소비량이 증가한다. 따라서 서버는 클라이언트의 요구 패턴에 따라 주문형 브로드캐스팅 채널의 대역폭을 적절히 조절해야 한다. 본 논문에서는 최대 주문형 브로드캐스팅 채널의 대역폭을 조

절함으로써, 논리적 채널의 대역폭을 조절하는 효과를 낸다.

클라이언트의 요구패턴에 따른 원인은 표 2와 같이 나타낼 수 있다.

표 2 클라이언트의 요구패턴

구분	데이터 종류 증가	데이터 종류 감소
클라이언트 요구 수 증가	주기적 브로드캐스팅 채널에서 만족하지 못하는 클라이언트가 많음	실제로 인기 있는 데이터가 일시적으로 브로드캐스트 되지 않음
클라이언트 요구 수 감소	소외(starvation) 발생	주기적 브로드캐스팅 채널에서 많은 클라이언트가 만족하고 있음

표 2에서 보는 바와 같이 클라이언트의 요구와 요구하는 데이터 종류가 증가하면 주기적 브로드캐스팅 채널에서 만족하지 못하는 클라이언트가 많음을 나타낸다. 이의 원인으로는 주문형 브로드캐스팅 채널의 대역폭이 작아져 소외된 데이터가 많이 발생했거나, 주기적 브로드캐스팅 채널의 대역폭이 작아져 실제 인기 있는 데이터가 방송되지 못했을 가능성이 있다. 따라서, 클라이언트의 요구와 데이터의 종류가 함께 증가할 경우는 요구 회수 증가율과 데이터 종류의 증가율을 비교하여 두 가지 경우로 나누어 그 원인을 파악하여 처리한다. 그리고 요구회수는 감소하지만 클라이언트가 요구하는 데이터의 종류가 증가하면 주문형 브로드캐스팅 채널의 대역폭을 늘려서 소외를 방지해야 한다. 표 2의 각 경우에 대한 처리방안은 표 3과 같다.

표 3 요구패턴에 따른 처리방안

구분	데이터 종류 증가	데이터 종류 감소
클라이언트 요구 수 증가	1) 데이터 종류의 증가율이 요구 수 증가율보다 크면, 주문형 브로드캐스팅 채널의 대역폭 증가 2) 데이터 종류의 증가율이 요구 수 증가율보다 작으면, 주문형 브로드캐스팅 채널의 대역폭 감소	그대로 유지
클라이언트 요구 수 감소	주문형 브로드캐스팅 채널의 대역폭 증가	그대로 유지

주문형 브로드캐스팅 채널에 대한 액세스 시간은 클라이언트의 요구회수보다 요구한 데이터의 종류에 더

많은 영향을 받는다. 데이터 종류와 요구회수가 함께 증가할 경우에는 증가율을 계산하여 요구회수의 증가율이 더 클 경우에는 주문형 브로드캐스팅 채널의 대역폭을 줄이고 그 반대의 경우에는 주문형 브로드캐스팅 채널의 대역폭을 늘려야 한다.

서버는 데이터 요구에 대한 정보를 저장한다. 정보 RI 는 $\langle Key, R \rangle$ 로 구성된다. Key 는 데이터를 구분하기 위한 키이고, R 은 현재까지의 해당 데이터에 대한 요구회수이다. 클라이언트의 요구가 처음 발생하면 요구 큐에 RI 정보를 추가한다. 이미 요청한 데이터에 대한 요구가 다시 발생하면 요구회수를 증가시킨다.

주문형 브로드캐스팅 채널의 대역폭을 결정하기 위해 필요한 요소는 요청된 데이터의 종류인 RI 의 튜플수와 서비스해야 할 전체 요구 수이다. n 번째 브로드캐스팅 주기에 RI 의 튜플수를 K_n , 전체 요구회수를 RQ_n 라 할 때, 데이터 종류 수와 요구회수의 증가율 k , r 과 주문형 브로드캐스팅 채널의 대역폭의 크기 ON_n 은 식 9로 결정된다.

$$k = \frac{|K_n - K_{n-1}|}{\max(K_{n-1}, K_n)}, \quad r = \frac{|RQ_n - RQ_{n-1}|}{\max(RQ_{n-1}, RQ_n)}$$

$$\begin{aligned} &\text{if } K_n > K_{n-1} \text{ and } RQ_n > RQ_{n-1}, \\ &\quad \text{if } r < k, \quad ON_n = ON_{n-1} \times (1+r) \times (1+k) \\ &\quad \text{if } r > k, \quad ON_n = ON_{n-1} \times (1-r) \times (1-k) \\ &\text{if } K_n > K_{n-1} \text{ and } RQ_n < RQ_{n-1}, \\ &\quad ON_n = ON_{n-1} \times (1+r) \times (1+k) \end{aligned} \quad (9)$$

데이터가 주문형 브로드캐스팅을 위해 선택되면 해당 튜플은 RI 에서 삭제된다. 주문형 브로드캐스팅 채널의 대역폭의 크기를 클라이언트의 요구패턴에 따라 동적으로 결정하면 일정한 최대 주문형 브로드캐스팅 채널의 대역폭의 크기를 이용할 때보다 액세스 시간과 전력 소비량 면에서 효율적이다. 성능평가에서 주문형 브로드캐스팅 채널의 대역폭의 동적 할당의 효과를 보일 것이다.

4. 성능평가

논문에서 제안한 데이터 크기를 고려한 브로드캐스팅 기법을 실험하기 위해 크기가 서로 다른 데이터를 생성하고 각 클라이언트의 요구에 따른 평균 액세스 시간, 평균 튜닝 시간, 평균 에너지 소비량을 구하여 데이터 크기를 고려하지 않은 기존의 하이브리드 브로드캐스팅 기법과 비교, 평가한다. 그리고 주문형 브로드캐스팅 채널의 대역폭의 동적 할당으로 인한 성능향상 정도를 평가한다.

4.1 실험 모델

실험 모델은 크게 데이터베이스 및 서버 부분, 클라이

언트 부분으로 나누어진다.

4.1.1 데이터베이스 및 서버

데이터베이스는 *DatabaseSize*개의 데이터 아이템으로 구성된다. 아이템의 크기인 *DataSize*는 *Database Size*의 80%는 작은 크기로 20%는 큰 크기로 할당된다. 각 데이터의 인기도는 *zipf-like distribution*을 따르도록 하였다. *zipf skew*는 0과 1 사이의 수이다. *zipf skew*에 따른 각 데이터 아이템의 인기도는 식 10과 같다.

$$P_i = \frac{1}{i^z} \sum_{j=1}^v \frac{1}{j^z} \quad (0 < P_i < 1) \quad (10)$$

인덱스 베킷은 <검색키, 포인터> 리스트로 구성된다. 키와 포인터는 각각 2 Byte라고 가정한다. 따라서 인덱스 트리의 팬아웃이 *IndexTreeFanout*일 때 인덱스 세그먼트의 한 베킷의 크기는 *IndexTreeFanout* × 4 Byte가 된다.

클라이언트의 데이터 요구 패킷은 <클라이언트 아이디, 키>로 구성된다. 아이디는 클라이언트를 구분하기 위한 유일한 아이디이며, 키는 요청하는 데이터의 키이다. 아이디는 클라이언트를 유일하게 구분하기 위해 16 Byte로 하고, 키는 2 Byte로 한다.

*Bcast*는 브로드캐스팅 주기로 *DatabaseSize*의 크기에 따라 결정된다. 최대 주문형 브로드캐스팅 채널의 대역폭의 크기는 동적으로 할당하지 않을 때는 *MaxOnDemand*으로 일정하다. 그리고 *Upstream* 채널의 대역폭은 *UpBandwidth*이고 *Downstream*의 대역폭은 *DownBandwidth*로 *UpBandwidth*는 *DownBandwidth*보다 훨씬 작다.

4.1.2 클라이언트

클라이언트의 요구(데이터 필요 발생)는 *Clients*개의 클라이언트에 의해 *ArrivalRate*의 시간 간격으로 발생한다. 본 논문에서는 λ 시간동안 셀에 머무르는 평균 클라이언트의 수를 100으로 하고, 한 클라이언트가 λ 시간동안 평균 100~200개의 요구를 하는 것으로 하여 실험한다. 요구 도착률은 포아송(poison) 분포를 따르도록 하였으며, 요구 패턴은 데이터의 온도와 마찬가지로 지프(*zipf-like*) 분포를 따른다. 논문의 브로드캐스팅 기법의 전력소모 절약 효과를 보여주기 위해 각 클라이언트 요구에 대한 전력 소비를 측정한다. 전력 소비량을 측정하기 위해서 클라이언트의 각 상태의 비용을 설정하였다. 모든 클라이언트는 같은 상태에서 같은 양의 전력을 소비한다고 가정한다. 그리고 디스크, CPU, 디스플레이에 소모되는 전력은 무시하고 이동 컴퓨터의 모뎀의 전력소비량만 고려하기로 한다.

휴면(Doze), 활동(Active), 요구(Request) 상태의 전력 소비량은 wireless PCMCIA modem에 기초하여[11], 각 상태에서 전력 소비량은 식 11로 계산된다.

$$\text{Energy(joules)} = \text{Power(Watts)} \times \text{Time(sec)}. \quad (11)$$

4.2 성능평가 척도와 실험에서 사용하는 변수

논문에서 제안하는 데이터 크기를 고려한 브로드캐스팅 기법의 목표는 클라이언트의 데이터 액세스 시간과 전력 소비량을 최소화하는 것이다. 평균 액세스 시간, 평균 에너지 소비량을 통해 제안한 기법과 크기를 고려하지 않은 기법의 성능을 비교한다. 또한 주문형 브로드캐스팅 채널의 대역폭의 동적 할당의 효과를 평균 액세스 시간과 전력 소비량 면에서 평가할 것이다.

표 4에 실험에서 사용하는 변수를 나타내었다.

표 4 시뮬레이션에 사용한 변수

	변수	값
서버	DatabaseSize	1000 items
	DataSize	0.5KB~50KB
	IndexTreeFanout	3
	Bcast	전체 데이터의 20%
	SimulationTime	250주기
	ReqPacketSize	18 Byte
클라이언트	λ	4000s~8000s
	ZipfSkew	0.1~1.0
	Clients	100
	ArrivalRate	1s~80s
	Doze, Active, Request	0.1, 0.2, 0.5 (watt/s) (Motorola wireless PCMCIA modem, '97)
전송 채널	DownBandwidth	32 Kbps
	UpBandwidth	0.5 Kbps
	MaxOnDemand	10%

4.3 실험 결과

4.2에서 밝힌 성능평가 척도에 따라 크기를 고려한 경우와 고려하지 않은 경우의 성능과 주문형 브로드캐스팅 채널의 대역폭의 동적 할당의 효과를 평가한다.

4.3.1 크기를 고려한 브로드캐스팅 기법의 성능

제안한 스케줄링 기법과 온도만 고려한 스케줄링 기법의 성능을 비교하기 위하여 데이터 크기를 다양하게 분포시키면서 시뮬레이션하였다. 이 실험에서는 데이터 크기를 고려하는 선택 방법의 효과를 보기 위해 최대 주문형 브로드캐스팅 채널의 대역폭의 크기는 20%로 고정한다. 처음 50 주기는 시스템의 적응을 위한 것으로, 이 주기의 액세스 시간, 튜닝 시간과 전력 소비량은 무시하였다.

그림 7, 그림 8, 그림 9는 zipf skew를 증가시키면서 클라이언트의 평균 액세스 시간, 평균 투닝 시간, 평균 전력소비량을 측정한 결과이다. 데이터 크기를 고려하지 않는 기존의 하이브리드 브로드캐스팅 기법에 비해 데이터 크기를 고려한 브로드캐스팅 기법이 좋은 성능을 나타내고 있다.

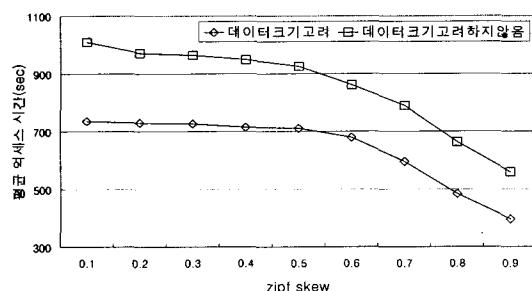


그림 7 평균 액세스 시간

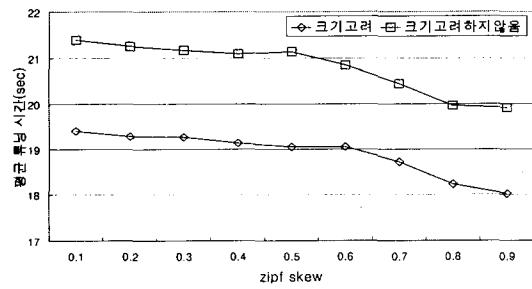


그림 8 평균 투닝 시간

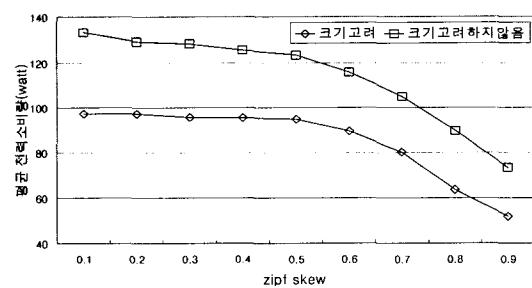


그림 9 평균 전력 소비량

한 결과 본 논문에서 제안하는 데이터 크기를 고려한 기법의 경우 약 180개(평균크기보다 작은 데이터는 약 170개)였고, 데이터 크기를 고려하지 않는 기존의 기법의 경우는 약 135개(평균크기보다 작은 데이터는 약 120개)였다. 즉, 이 논문에서 제안한 브로드캐스팅 기법은 데이터의 온도, 즉, $TF(i)$ 값이 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. 실험 결과의 모든 채널에서 데이터 크기를 고려한 경우가 더 나은 성능을 보였다.

서비스 클라이언트의 요구를 받으면 λ 시간 내에 적어도 한번 서비스해야 한다. 실험에서 λ 시간을 지키지 못하는 회수를 측정한 결과 크기를 고려한 경우와 고려하지 않은 경우 모두 λ 시간을 지키지 못하는 회수는 0이었다. λ 시간은 데이터의 온도와 주문형 브로드캐스팅 데이터를 선택하는데 필요한 IF에 의해 지켜질 수 있었다.

4.3.2 주문형 브로드캐스팅 채널의 대역폭의 동적인 할당에 따른 성능

4.3.1에서는 최대 주문형 브로드캐스팅 채널의 대역폭을 일정하게 하여 성능을 측정하였다. 이 절에서는 주문형 브로드캐스팅 채널의 대역폭을 동적으로 할당할 경우의 성능을 측정하고 크기를 일정하게 한 경우와 비교한다.

그림 10, 그림 11은 주문형 채널의 대역폭의 크기를 동적으로 할당 할 때 zipf skew를 증가시키면서 클라이언트의 평균 액세스 시간, 평균 전력소비량을 측정한 결과이다. 전반적으로 주문형 브로드캐스팅 채널의 대역폭의 크기를 동적으로 할당할 때 성능이 조금 더 좋았거나, 평균 액세스 시간은 0.2%~5%까지 성능이 향상되었고, zipf skew가 증가할수록 즉, 클라이언트의 요구가 특정 데이터에 집중할수록 클라이언트의 요구 패턴이 일정해지므로 성능 향상 정도가 감소하였다. 평균 전력

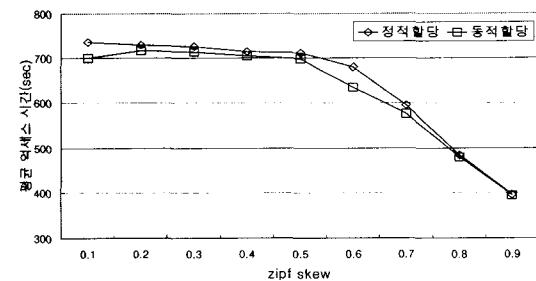


그림 10 평균 액세스 시간

실험에서 한 주기에 할당되는 평균 데이터 수를 측정

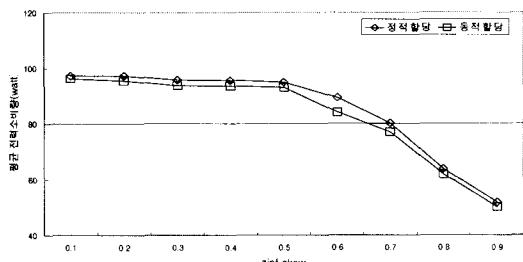


그림 11 평균 전력소비량

소비량은 1%~5% 가량 성능이 향상되었다.

5. 결론 및 향후연구

브로드캐스팅은 비대칭적 통신 환경을 가진 이동 컴퓨팅 환경에 유용하다. 브로드캐스팅은 확장성이 있으며 많은 클라이언트에 정보를 전달할 수 있다. 이동 클라이언트에 있어서 중요한 성능 요소는 데이터 액세스 시간과 전력 소비량이라고 할 수 있다. 본 논문에서는 클라이언트의 데이터 액세스 시간을 줄이고 전력을 절약할 수 있는 적응적 브로드캐스팅 기법을 제안하였다. 데이터 액세스 시간과 전력 소비량은 브로드캐스트할 데이터의 선택 방법과 인덱싱 기법에 달려있다.

기존의 많은 논문들이 서버가 전달하는 데이터의 크기가 같다고 가정하고 스케줄링 알고리즘을 제안하였다. 그렇지만 실제로는 데이터의 크기는 다양하다. 그리고 대부분의 연구에서 클라이언트의 이동성을 고려하지 않았다. 그러나 이 논문에서는 주기적 브로드캐스팅 데이터를 선택하기 위해 데이터의 온도 요소와 크기 요소, 이동성을 고려한 기법을 제안하였다. 데이터의 온도 요소는 클라이언트의 요구와 클라이언트의 이동성을 바탕으로 예측된다. 그리고 데이터의 크기 요소는 최대 크기에 대한 상대적 비율이다. 각 데이터의 온도 요소와 크기 요소를 통해 우선순위 값을 계산한 후 주기적 브로드캐스팅 데이터를 선택한다. 주기적 브로드캐스팅 채널에서 만족하지 못한 클라이언트의 요구는 주문형 브로드캐스팅 채널의 대역폭의 크기에도 영향을 받는다. 본 논문에서는 클라이언트의 요구를 더 효율적으로 서비스하기 위해 주문형 브로드캐스팅 채널의 대역폭의 크기를 동적으로 할당하는 방법을 제시하고, 성능 평가를 통해 동

적으로 할당했을 때의 효과를 보였다. 클라이언트 요구 회수와 요구된 데이터 종류의 증감을 통해 클라이언트의 액세스 패턴을 예측하고 그에 따라 주문형 브로드캐스팅 채널의 대역폭을 조정한다. 주문형 브로드캐스팅 채널의 대역폭을 동적으로 조정할 때 클라이언트의 액세스 시간과 전력소비량이 감소하였다. 클라이언트의 액세스가 특정 데이터로 집중될수록 클라이언트의 요구패턴이 일정해져서 그 효과는 감소하였다.

본 논문에서는 브로드캐스트할 데이터를 선택한 후 클라이언트에게 데이터가 도착할 시간을 알려주기 위한 인덱싱 기법으로 (1:m) 인덱싱을 이용하였다. (1:m) 인덱싱 기법은 데이터베이스에 있는 모든 데이터를 전송한다고 가정한 방식이다. 게다가 데이터의 인기도를 고려하지 않고 데이터를 키 순서로 배치하였다. 이 인덱싱 방법은 데이터베이스의 일부 데이터만 전달하거나 클라이언트의 직접적인 요청이 있는 하이브리드 방식과 맞지 않다. 따라서 하이브리드 방식에 맞는 새로운 인덱싱 기법에 대한 연구가 이루어져야 한다. 또한, 본 논문에서 제안하는 브로드캐스팅 기법은 대부분의 기존 연구에서와 마찬가지로 신뢰성 있는 전송 채널을 가정하고 있다. 그러나, 이동 컴퓨팅 환경에서는 전송 채널의 신뢰성을 보장하지 못하므로, 향후 연구에서는 전송 채널의 신뢰성을 고려하는 브로드캐스팅 기법에 대해 연구할 것이다.

참고 문헌

- [1] S. Acharya, R. Alonso, M. Franklin and S. Zdonik, "Broadcast Disks: Data Management for Asymmetric Communications Environments," SIGMOD Conference, 199-210, 1995.
- [2] P. Xuan, S. Sen, O. Gonzalez, J. Fernandes and K. Ramamritham, "Broadcast on demand: Efficient and Timely Dissemination of Data in Mobile Environments," Real-Time Technology and Applications Symposium, 38-48, 1997.
- [3] D. Aksoy and M. Franklin, "Scheduling for Large-Scale On-Demand Data Broadcasting," INFOCOM '98 vol.2 651 -659 1998.
- [4] A. Datta, "Broadcast Protocols to Support Efficient Retrieval from Databases by Mobile Users," ACM Transactions on Database Systems, Vol. 24, No. 1, 1-79, 1999.
- [5] Hu, J.H. "A Novel Push-and-Pull Hybrid Data Broadcast Scheme for Wireless Information Networks," ICC 2000, vol.3, 1778 -1782, 2000.
- [6] K. Stathatos, N. Roussopoulos and J.S. Baras, "Adaptive Data Broadcast in Hybrid Networks," VLDB, 326-335, 1997.

- [7] J.H. Oh, K.A. Hua and K. Prabhakara, "A New Broadcasting Technique for An Adaptive Hybrid Data Delivery in Wireless Mobile Network Environment," Proc. of 19th IEEE International Performance, Computing and Communications Conference, 361-367, 2000.
- [8] J. Fernandez and K. Ramamritham, "Adaptive Dissemination of Data in Time-Critical Asymmetric Communication Environments," Proceedings of 11th Euromicro Conference on Real-Time Systems, 1999.
- [9] R. Agrawal and P. K. Chrysanthis, "Efficient Data Dissemination to Mobile Clients in E-Commerce Applications," WECWIS, 2001.
- [10] T. Imielinski, S. Viswanathan and B.R. Badrinath, "Data on Air: Organization and Access," IEEE Transactions on Knowledge and Data Engineering, Vol 9, NO. 3, 1997.
- [11] Motorola, "Personal messenger 100c model specifications. <http://www.mot.com>," 1997.
- [12] C.J. Su and L. Tassiulas, "Designing Broadcast Schedules for Information Dissemination through Broadcasting," TR. <http://www.isr.umd.edu/CSHCN/>, 1997.
- [13] D. Barbara, "Mobile Computing and Databases-A Survey," IEEE Transaction on Knowledge and Data Engineering, Vol. 11, 108-117, 1999.
- [14] K.Y. Lam, E. Chan and J.H. Yuen, "Approaches for Broadcasting Temporal Data in Mobile Computing Systems," The Journal of Systems and Software.



김 경 석

1977년 서울대학교 무역학과(경제학사)
1979년 서울대학교 전자계산학과(이학석사). 1988년 일리노이 주립대(어바나-샴페인) 전자계산학 박사. 1988년~1992년 미국 노스다코타 주립대학교 전자계산학과 조교수. 1992년~현재 부산대학교 전자전기정보컴퓨터공학부 교수. 관심분야는 데이터베이스, 멀티미디어, 한글/한말 정보처리



유 영 호

1994년 부산대학교 전자계산학과(이학사)
1997년 부산대학교 전자계산학과(이학석사). 1997년~현재 부산대학교 전자계산학과 박사과정. 관심분야는 데이터베이스 XML, 이동 통신



이 종 환

1996년 부산대학교 전자계산학과(이학사)
1998년 부산대학교 전자계산학과(이학석사)
1998년~현재 부산대학교 전자계산학과 박사과정. 관심분야는 데이터베이스, 정보가전, XML