

이동 컴퓨팅 환경에서 위치 의존 질의에 적합한 선형 클러스터링을 이용한 브로드캐스팅 기법

°정 일동, 유 영호, 이 중화, 신 지현, 김 경석
부산대학교 전자계산학과
(idjung, yhyou, jhlee, jhshin, gimgs}@asadal.cs.pusan.ac.kr

Using Linear Clustering for Broadcasting to support Location Dependent Query in Mobile Computing Environment

°Il-dong Jung Young-ho Yu, Jung-hwa Lee, Ji-hyun Shin, Kyongsok Kim
Department of Computer Science, Pusan National University

요 약

이동 컴퓨팅 환경에서 이동 호스트의 위치가 변함에 따라 그 의미가 달라지는 위치 의존 질의를 효과적으로 처리하기 위해서는 이동 호스트의 캐시 기법이 중요하지만, 위치 의존 질의를 효과적으로 지원할 수 있는 지구국의 브로드캐스팅 기법도 중요하다. 본 논문에서는 지구국이 담당하는 영역을 격자로 나누어 인덱싱한 데이터를 위치 의존 질의에 적합하도록 공간-채움 곡선을 이용해서 선형 클러스터링하여 구성 시간을 줄이는 브로드캐스팅 기법을 제안하고, 구성 시간을 포함시킨 이동 호스트의 활동 시간을 측정하여 그 성능을 비교한다.

1. 서론

무선 통신 기술과 휴대형 정보 장치의 발달로 등장한 이동 컴퓨팅 환경 (Mobile Computing Environment) 은 사용자가 랩탑이나 PDA와 같은 휴대 가능한 장비를 이용해서 사용자의 물리적인 위치나 이동에 상관없이 무선 통신을 이용해서 서버 혹은 다른 컴퓨터의 자원과 함께 작업하는 것을 말한다. 이동 컴퓨팅 환경은 전통적인 컴퓨팅 환경과 달리 무선 통신의 낮은 대역폭, 잦은 접속 단절, 호스트의 이동성, 배터리 용량 등과 같은 제약이 있다[7].

이동 컴퓨팅 환경의 제약 사항을 고려하여 이동 호스트에서 자주 요구하는 데이터를 지구국에서 브로드캐스팅해서 전달함으로써 이동 호스트가 데이터를 요구하는데 필요한 배터리 사용을 줄이고 지구국이 담당하는 영역 안에 이동 호스트의 수가 증가하더라도 효과적으로 처리할 수 있다[4].

최근 이동 컴퓨팅 환경에서 보편적인 형태가 되고 있는 위치 의존 질의 (Location Dependent Query) 는 위치에 의존하는 데이터를 처리하는 질의이다. 위치 의존 질의는 질의의 결과를 만들어 내는 중요한 척도가 위치이다[9]. 위치 의존 질의를 효과적으로 처리하기 위해서는 이동 호스트의 캐시가 중요하지만, 지구국이 위치 의존 질의를 효과적으로 지원하기 위한 브로드캐스팅 기법도 중요하다. 지구국이 브로드캐스팅으로 어떤 기법을 사용하는가에 따라 이동 호스트의 대기 모드와 활동 모드의 전환 횟수가 줄어들어 전력을 많이 소모하는 구성 시간과 활동 시간이 달라진다.

본 논문에서는 위치 의존 질의에서 처리하는 데이터를 선형 클러스터링하여 브로드캐스팅하는 기법을 제안하고, 이동 호스트의 구성 시간 (setup time) 을 측정해서 브로드캐스팅 기법의 성능을 평가한다.

2. 관련 연구

2.1 위치 의존 질의

위치 의존 질의는 이동 호스트의 위치에 따라 그 의미가 바뀌는 질의를 말한다. 보기를 들어 충청북도 지방을 여행 중

인 사용자가 "가장 가까운 호텔을 찾아라."라는 질의를 요구하면 이동 호스트는 현재 위치에서 가장 가까운 호텔인 "A"의 위치를 질의 결과로 돌려주고, 경상남도에서 같은 질의를 하였다면 이동 호스트의 현재 위치에서 가장 가까운 호텔인 "B"의 위치를 질의 결과로 돌려준다. 보기에서 위치 의존 질의는 다음과 같은 특징을 가짐을 알 수 있다.

- 1) 데이터의 의미와 가치가 특정 영역과 연관되어 있다.
- 2) 질의의 결과가 질의를 요구한 이동 호스트의 지리학적 위치에 따라 달라진다.
- 3) 질의 결과는 이동 호스트의 현재 위치에서만 의미를 가진다[3].

현재까지 이동 컴퓨팅 환경에서 위치 의존 질의에 관한 연구는 위치 의존 데이터에 관한 연구[3], 위치 의존 질의를 처리하는 이동 호스트의 캐시 기법에 관한 연구를 중심으로 이루어졌다[8][9].

이동 호스트의 캐시 기법만으로는 이동 컴퓨팅 환경에서 우수한 품질의 서비스를 만들 수 없고, 브로드캐스팅 기법의 지원이 있어야 한다. 본 논문에서는 이동 호스트의 위치 의존 질의를 효과적으로 지원할 수 있는 브로드캐스팅 기법을 제안한다.

2.2 브로드캐스팅 기법

브로드캐스팅 기법을 이용해서 이동 호스트의 전력 소모량을 줄이기 위한 연구는 많다. [5]는 브로드캐스팅 1주기 동안 인덱스를 여러 번 브로드캐스팅하여 조율 시간 (tuning time) 을 줄이는 (1,m) 인덱싱 기법과 분산 인덱싱 기법을 제안하고, 제안한 알고리즘으로 이동 호스트의 전력 소모를 줄일 수 있음을 입증했다. 최근 연구 중 [2]는 인기도와 무시 (ignore) 정도를 이용해서 우선 순위를 정하고 우선 순위에 따라 브로드캐스팅할 데이터를 결정하는 적응형 프로토콜을 제안하였다. 적응형 프로토콜에서는 이동 호스트의 접근 시간과 조율 시간을 최소로 하여 이동 호스트의 전력 소모에 대한 고려를 하고 있다.

따라서 본 논문에서는 이동 호스트의 전력 소모를 최소화할 수 있도록 위치 의존 질의에서 사용하는 데이터를 선형 클

러스터링하는 브로드캐스팅 기법을 제안한다.

3. 위치 의존 질의를 지원하는 브로드캐스팅 기법

3.1 가정과 정의

본 논문에서는 [1]에서 제안한 이동 컴퓨팅 환경의 모델을 사용하며, 다음 몇 가지를 가정하고 있다.

- 1) 한 지구국은 자신이 담당하는 셀 영역 내의 이동 호스트들에게 자신의 데이터베이스 내에 있는 데이터를 주기적으로 브로드캐스팅한다.
- 2) 이동 호스트는 지구국의 셀 내에서 항상 자신의 위치를 알 수 있다.
- 3) 이동 호스트의 캐시 용량은 서버의 데이터베이스의 용량보다 작고, 이동 호스트는 브로드캐스팅되는 내용 중 자신이 필요한 것을 선택적으로 캐시에 저장할 수 있다.
- 4) 이동 호스트는 이동 호스트의 캐시 영역과 겹치는 데이터를 캐시에 저장한다[8].

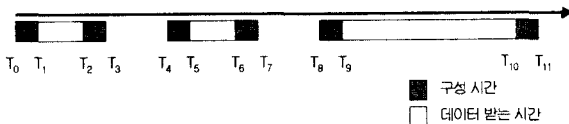
[정의 1] 캐시 영역 (Cache Region) 은 이동 호스트의 캐시가 담당하는 영역이다. 즉, 캐시에 저장되는 영역이다. 크기는 캐시에 저장되는 격자 수로 표시한다.

[정의 2] 조율 시간 (Tuning Time) 은 이동 호스트가 원하는 데이터를 받기 위해서 채널에 연결하는 시간이다.

[정의 3] 구성 시간 (Setup Time) 은 이동 호스트가 대기 모드 (doze mode) 에서 활동 모드 (active mode) 로 바꾸거나, 활동 모드에서 대기 모드로 바꾸는데 걸리는 시간이다[5].

[정의 4] 활동 시간 (active time) 은 이동 호스트의 전력 소모 시간의 대부분을 차지하는 활동 모드에서 이동 호스트가 작동한 시간과 구성 시간의 합으로 표시한다.

의미를 규정한 단어를 [그림 1]에서 보기를 들어 알아보겠다. 이동 호스트가 $(T_2 - T_1)$ 시간 동안 인덱스를 받고, $(T_6 - T_5)$ 시간과 $(T_{10} - T_9)$ 시간 동안 원하는 데이터를 받았다면, 조율 시간 = $(T_2 - T_1)$ 이다. 이동 호스트가 대기 모드일 때 채널에서 데이터를 바로 받을 수 없으므로 활동 모드로 바꾸는 시간과 활동 모드에서 대기 모드로 바꾸는 시간이 필요하며, 구성 시간 = $(T_{11} - T_{10}) + (T_9 - T_8) + (T_7 - T_6) + (T_5 - T_4) + (T_3 - T_2) + (T_1 - T_0)$ 이다.



[그림 1] 조율 시간과 구성 시간

이동 호스트의 캐시 영역의 한계로 인해서 이동 호스트가 지구국으로부터 받을 수 있는 데이터 수는 고정되어 있기 때문에 데이터를 받는 시간을 줄일 수는 없다. 본 논문에서는 데이터를 인접해서 브로드캐스팅하여 $(T_9 - T_8)$ 시간과 $(T_7 - T_6)$ 시간을 줄임으로써 구성 시간을 줄일 수 있는 브로드캐스팅 기법을 제안한다.

3.2 위치 의존 질의를 지원하는 브로드캐스팅

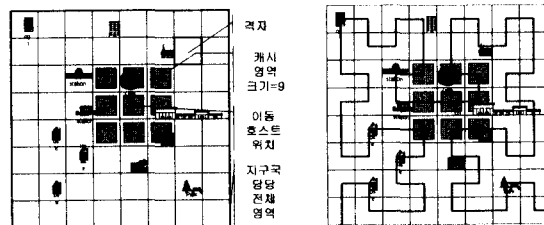
이동 컴퓨팅 환경에서 지구국이 데이터를 브로드캐스팅할 때, 이동 호스트의 조율 시간 (Tuning Time) 을 줄이기 위해서 지구국은 브로드캐스팅 순서를 알려주는 인덱스를 먼저 브로드캐스팅한다.

데이터는 지리 데이터이므로 2차원 공간에 분포하는 것으로 가정한다. 2차원 공간인 전체 영역에 존재하는 데이터를

브로드캐스팅하기 위해서 데이터의 브로드캐스팅 순서를 정해야 한다. 데이터의 좌표를 이용해서 브로드캐스팅 순서를 결정하면 각 데이터의 순서에 일정한 규칙이 없고 데이터가 흩어져서 브로드캐스팅된다.

본 논문에서는 지구국이 담당하는 전체 영역을 [그림 2]와 같이 격자 (grid) 로 나누어서 격자 별로 데이터를 저장하고 격자 단위로 브로드캐스팅하도록 한다. 격자 단위로 인덱스를 구성하므로 전체 인덱스의 크기가 작아지므로 인덱스의 브로드캐스팅 시간이 짧아진다. 따라서 이동 호스트의 조율 시간이 짧아진다.

이동 호스트는 자신의 캐시 영역에 속하는 데이터를 저장하기 때문에 이동 호스트에서 필요한 데이터가 인접해서 브로드캐스팅되면 이동 호스트는 구성 (setup) 횟수가 줄어들어 활동 시간이 줄어든다. 보기를 들어 캐시 영역의 크기가 9인 이동 호스트가 [그림 2]와 같이 (5,4) 격자에 위치하고 인접한 격자가 [그림 1]의 $(T_6 - T_5)$ 시간과 $(T_{10} - T_9)$ 시간 동안 브로드캐스팅된다면 이동 호스트는 데이터를 받기 위해서 데이터가 브로드캐스팅되기 전에 활동 모드로 바꾼 후 데이터를 기다려야 한다. 만약 두 데이터가 인접해서 브로드캐스팅되면 한 번만 구성해서 두 데이터를 받을 수 있으므로 $(T_7 - T_6)$ 시간과 $(T_9 - T_8)$ 시간이 필요하지 않다.



[그림 2] 영역을 격자로 나눔 [그림 3] 공간-채움 곡선 적용

이와 같은 방법으로 [그림 2]의 인접한 격자를 클러스터링해서 데이터를 브로드캐스팅하면 이동 호스트의 구성 시간이 줄어든다. 하지만, 이동 호스트가 전체 영역에 분포하고 있다면 인접 격자가 인접해서 브로드캐스팅되는 특정 격자에 속한 이동 호스트를 제외한 나머지 이동 호스트의 구성 시간이 커진다. 그러므로 대부분의 이동 호스트가 좋은 성능을 얻을 수 있는 방법이 필요하다.

본 논문에서는 인접한 공간 객체를 클러스터링해서 1차원으로 치러할 때 사용하는 선형 클러스터링 방법을 사용해서 브로드캐스팅 순서를 결정할 것을 제안한다.

[그림 3]과 같이 공간-채움 곡선 (Space-filling Curve) 을 이용해서 전체 지역의 방송 순서를 정한다. 공간-채움 곡선을 사용하여 클러스터링했기 때문에 전체 영역의 격자가 인접해서 브로드캐스팅된다. 그러므로 데이터를 기다리는 이동 호스트의 구성 (setup) 횟수가 줄어든다. 따라서, 이동 호스트의 전체 사용 시간 중에서 구성 시간이 줄어들기 때문에 전력을 효율적으로 사용할 수 있다.

4. 실험

4.1 실험 방법

이동 호스트의 전력 소모량은 활동 시간이 중요한 인자이다. 전력 소모량은 각 환경마다 다르기 때문에 활동 시간을 계산해서 예측해 볼 수 있다. 본 논문에서는 여러 가지 공간-채움 곡선에 따른 브로드캐스팅을 하고 그에 따른 활동 시간을 측정해서 브로드캐스팅 기법의 성능을 평가한다. 실험에서 사용한 인자 값은 [표 1]과 같다.

clustering 인자는 브로드캐스팅 순서를 결정할 때 사용하는 공간-채움 곡선을 나타낸다. 공간-채움 곡선으로는 일반적인

으로 Row-wise, Z-Order, Hilbert, Gray code 곡선을 많이 사용한다[6]. 본 논문에서는 Hilbert, Z-Order, Row-wise 곡선을 비교한다.

실험에서 사용한 이동 호스트의 위치와 수는 전체 영역에 고르게 분포할 수 있도록 생성한다.

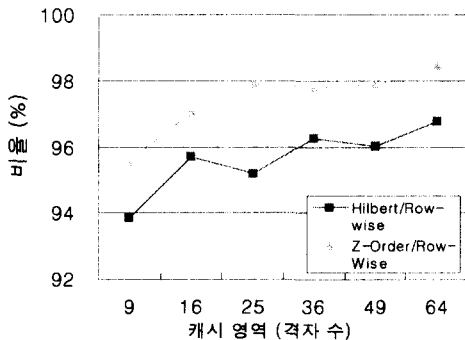
인자	설명	값
region	전체 영역	3.2 (Km) * 3.2 (Km)
grid size	격자 크기	100 (m) * 100 (m)
data size	격자 내의 데이터	3 * 128 (bytes)
period	브로드캐스팅 주기	54.85 (초)
data	브로드캐스팅 단위	128 (bytes)
clustering	브로드캐스팅 순서 결정 방법과 클러스터링	row-wise, z-order, hilbert
bandwidth	무선 네트워크 대역폭	56 (Kbps)
cache region	캐시 영역의 크기	9, 16, 25, 36, 49, 64
setup time	구성 시간	0.02 (초)

[표 1] 실험 인자 값

위치 의존 질의는 이동 호스트의 캐시 영역의 데이터를 사용해서 처리된다. 질의 대상이 되는 영역이 커짐에 따라 여러 가지 브로드캐스팅 기법의 적응성을 평가하기 위해 질의 대상을 저장하는 캐시 영역의 크기를 변화시키며 실험한다.

4.2 실험 결과

이동 호스트의 활동 시간이 클러스터링의 지표가 될 수 있기 때문에 활동 시간을 평가 지표로 사용한다. [그림 4]는 Row-wise의 순서로 브로드캐스팅했을 경우의 이동 호스트의 활동 시간에 대한 Hilbert와 Z-Order의 순서로 브로드캐스팅했을 때 이동 호스트의 활동 시간의 비율을 표시한 그래프이다.



[그림 4] 이동 호스트의 평균 활동 시간 비율

Row-wise 곡선은 캐시 영역의 가로 혹은 세로의 격자 1개가 증가할 때마다 구성 시간이 setup time만큼 늘어나는데 비해서, 다른 곡선은 구성 시간이 선형으로 증가하지 않기 때문에 [그림 4]의 활동 시간의 비율이 일정하지 않다.

[그림 4]에서 격자 수가 25일 때 Hilbert 곡선과 Z-Order 곡선의 차이가 많이 난다. 임의의 5*5 영역에서 각 인접 격자 간 평균 거리가 Hilbert 곡선을 사용한 경우보다 Z-Order 곡선을 사용한 경우가 더 크게 나타나는데, 그로 인해 구성 시간이 많아진 것으로 보인다.

캐시 영역이 넓어지면 어떤 브로드캐스팅 방법을 사용하더라도 인접한 격자의 의미가 없어지기 때문에 Row-wise 곡선의 활동 시간과 Hilbert 곡선과 Z-Order 곡선의 활동 시간이 비슷하게 나타나는 것으로 보인다.

실험 결과에서 Row-wise 곡선을 사용한 경우에 비해 Hilbert 곡선을 사용한 경우에 전반적으로 좋은 성능을 보인다. Row-wise 곡선은 $\sqrt{(cache\ region)}$ 번 클러스터링되어 브로드캐스팅되며, $\sqrt{(cache\ region)} * setup\ time$ 만큼의 구성 시간이 필요하다. Z-Order 곡선과 Hilbert 곡선은 캐시 영역의 크기마다 다르겠지만 Row-wise 곡선에 비해서 적은 횟수로 클러스터링되어 브로드캐스팅되기 때문에 구성 시간이 적게 필요하다.

5. 결론 및 향후 연구 과제

본 논문에서는 데이터를 선형 클러스터링하여 위치 의존 질의에 적합한 브로드캐스팅 기법을 제안하였다. 이동 호스트의 캐시 영역의 크기를 변화시키면서 여러 가지 선형 클러스터링 방법의 캐시 영역에 대한 적응성을 실험하였다. 그리고 이동 호스트의 활동 시간을 이용해서 여러 가지 선형 클러스터링에 따른 브로드캐스팅 기법의 성능을 비교하였다. 제안한 브로드캐스팅 기법은 다양한 크기의 캐시 영역에서 잘 적응하고, 이동 호스트의 활동 시간이 줄어들어 이동 호스트의 전력 소모를 줄일 수 있음을 보였다.

본 논문에서는 2*2 격자를 기본으로 하는 공간-채움 곡선을 사용했는데, 각 캐시 영역의 크기에 잘 적응할 수 있는 변형 공간-채움 곡선에 대한 연구를 통해서 이동 호스트의 특징을 잘 지원할 수 있는 브로드캐스팅 기법의 연구가 필요하다. 그리고, 본 논문은 위치 의존 질의를 효과적으로 처리하는 이동 호스트의 캐시를 효과적으로 지원하기 위한 연구이므로 이후에 캐시 정책에 대한 연구도 역시 필요하다.

6. 참고 문헌

- [1] D. Barbara, T. Imielinski, "Sleepers and Workaholics: Caching Strategies in mobile Environment", ACM SIGMOD '94, pp. 1~12, 1994
- [2] A. Datta, D.E. Vandermeer, A. Celik, V. Kumar, "Broadcast Protocols to Support Efficient Retrieval from Databases by Mobile Users", ACM Transactions on Database Systems, Vol. 24, No. 1, 1999
- [3] Margaret H. Dunham, Vijay Kumar, "Location Dependent Data and its Management in Mobile Databases", Mobility in Database and Distributed System, 1998
- [4] T. Imielinski, S. Viswanthan, "Wireless Publishing: Issues and Solutions", Mobile Computing, Kluwer Academic Publishers, pp. 299~329, 1996
- [5] T. Imielinski, S. Viswanthan, B.R. Badrinath, "Data on Air: Organization and Access", IEEE Transactions on Knowledge and Data Engineering, Vol 9, No 3, pp. 353~372, 1997
- [6] H. V. Jagadish, "Linear clustering of objects with multiple attributes", ACM SIGMOD '90, pp. 332~342, 1990
- [7] Jin Jing, Abdelsalam Helal, Ahmed Elmagarmid, "Client-Server Computing in Mobile Environments", ACM Computing Surveys, Vol. 31, No. 2, pp. 117~157, 1999
- [8] 김 호숙, 용 환승, "이동 데이터베이스 시스템에서 데이터의 위치와 영역 특성을 고려한 캐쉬 교체 기법", 정보과학회논문지: 데이터베이스 제 27권 제 1호, pp. 53~63, 2000
- [9] Qun Ren, Margaret H. Dunham, "Using Semantic Caching to Manage Location Dependent Data in Mobile Computing", ACM MobiCom '00, pp. 210~221, 2000