DSMS에서 영역을 포함하는 공간 연속질의 처리를 위한 R-tree기반의 집계기법

김상기[1], 이연[1], 이동욱[1], 오영환[2], 배해영[1]
[1]인하대학교 컴퓨터 정보 공학과
[2]나사렛대학교 정보통신학과
{kimsk, leeyeon, dwlee}@dblab.inha.ac.kr, yhoh@kornu.ac.kr, hybae@inha.ac.kr

Aggregation Method using R-tree for Spatial Continuous Query in DSMS

Sang-Ki Kim[1], Yan Li[1], Dong-Wook Lee[1], Young-Hwan Oh[2], Hae-Young Bae[1]
[1]Dept. of Computer Science and Information Engineering, Inha University
[2]Dept. of Information Science, Korea Nazarene University

요 약

DSMS는 USN과 같은 환경으로부터 스트림데이터를 실시간으로 입력 받아 등록된 연속질의를 처리하는 시스템이다. DSMS는 등록된 연속질의 처리를 위해 필요한 데이터를 버퍼에 관리하며, 스트림데이터의 저장기법에 따라 연속질의 처리 성능 및 버퍼 저장비용이 개선될 수 있으며, DSMS에서 연속질의는 특정 스트림데이터에 대해 일정한 기간 동안의 평균 값, 최대·소 값, 누적 값 등의 집계 연산을 요구하는 경우가 많다. 기존의 DSMS에서는 이러한 집계 연산이 필요한 연속질의의 효율적인 처리를 위해 LINT, BINT등의 자원 공유 집계 처리기법이 제안 되었다. 하지만 기존의 자원공유 집계 기법들은 위치 값을 포함하는 GeoSensing 데이터에 대한 고려를 하지 않았다. 본 논문에서는 공간 DSMS에서 공간영역질의 기반의 연속질의를 효율적으로 처리하기 위한 R-tree기반의 집계기법을 제안한다. 이는 각각의 연속질의에 포함된 공간 영역을 R-tree 인덱스로 구성하고, 연속질의에 필요한 공간 스트림데이터에 대한 집계 값을 저장하여 연속질의를 처리하는 것이다. 제안기법은 공간 DSMS에서 공간영역 기반의 연속질의 처리 성능을 개선할 수 있으며, R-tree 기반으로 해당 영역에 대한 데이터 만을 버퍼에 관리하여 저장비용을 줄일 수 있다.

1. 서 론

최근 유비쿼터스 환경에서 사용자 컨텍스트 기반의 실시간 서비스를 위해 USN(Ubiquitous Sensor Network), RFID 등에서 전송되는 스트림데이터를 처리하기 위한 DSMS(DataStream Management System)에 대한 연구가 활발히 진행 중이다.[1] 전통적인 데이터베이스 시스템과 다르게 DSMS는 처리 대상 데이터가 스트림 형태로 실시간, 연속적으로 발생하여 DSMS에 전송되는 모든 스트림데이터를 관리 및 저장하여 처리하기가 어렵다.[2] 이러한 DSMS(Data Stream Management System)는 빠르고 연속적인 스트림 데이터를 처리하기 위해 SQL에 기반한 연속질의(CQL: Continuous Query Language)를 사용하고 데이터를 관리하기 위한 메모리 관리 기법에 초점을 맞추고 있다.[3, 4] 연속질의는 여러 개의 질의가 등록되며, DSMS에서는 다수의 연속질의를 동시에 처리해야 하기 때문에 각 질의가 소유하는 독립적인 큐를 유지하여 중복 데이터에 대한 메모리 비용의 증가 및 이를 실시간으로 처리하는 부하가 따르는 문제가 발생한다. 따라서 기존의 DSMS에서는 연속질의 처리에 필요한 중복 데이터를 효율적으로 관리하기 위한 기존의 자원공유 기법이 연구되었다.[5]

DSMS에서 기존의 자원공유 기법은 BINT와 LINT, 그리고 단일 팬 구조를 이용하는 방법이 제안되었다.[6, 7] BINT는 저장된 최하단의 입력된 두 개의 튜플들로 집계 값을 계산하는 것을 시작으로 집계정보를 최상위단까지 계산한다. 이는 각각의 질의에 대해 독립적인 큐를 유지하지 않아도 되는 장점이 있다. 하지만 계층별 집계정보의 저장으로 메모리 유지 비용이 증가하고 질의 범위를 만족하는 집계 정보의 검색으로 인한 질의 처리 속도가 감소한다. 이러한 단점을 보완하기 위한 LINT는 집계 정보를 좌우 대칭으로 저장 관리를 하기 때문에 집계정보

본 연구는 건설교통부 첨단도시기술개발사업 - 지능형국토정보기술혁신 사업과제의 연구비지원(07국토정보C05)에 의해 수행되었습니다.

검색시간을 줄였다. 하지만 집계 정보를 유지하는 메모리 비용이 커지는 문제가 있다. BINT는 질의의 범위와 개수에 따라 최하단의 튜플과 상단의 집계정보를 동적으로 유지하기 때문에 메모리를 효율적으로 사용할 수 있지만, 공간 스트림데이터에 대한 연속질의의 경우 공간 영역을 기준으로 집계 값을 관리해야 하기 때문에 BINT와 같은 계층적 구조로 표현되기 어렵다.

기존의 자원공유 기법들은 비공간 스트림데이터만을 고려하였다. 그러나 유비쿼터스 환경에서의 다수의 응용 서비스는 위치 데이터를 포함하는 GeoSensor 정보를 요구한다. 예를 들면, "특정 지역의 1시간 동안 평균 온도", "특정 지역의 풍속" 등의 공간 영역 기반의 연속질의가 등록될 수 있다. 또한 이러한 공간 연속질의는 직접적인 데이터 보다는 평균이나 최대치와 같은 집계(Aggregation) 연산을 요구한다.[8] 본 논문에서는 위와 같은 공간 DSMS에서의 효율적인 연속질의 처리를 위해 R-tree 기반의 집계기법을 제안한다.[9] R-tree를 통해 질의를 인덱싱하고 단말 노드에 집계 값을 갱신하면서 유지 하기 때문에 다중 공간 질의 처리에 효과적이다.

본 논문의 구성은 다음과 같다. 2장에서는 기존 데이터 스트림 환경에서 자원공유 기법을 소개하고, 3장에서는 본 논문에서 제안한 공간 DSMS에서 효율적인 공간영역질의 처리를 위한 R-tree 기반의 집계 기법을 논의하고, 4장에서는 성능평가 결과를 제시하고, 5장에서는 결론을 맺고 향후 연구를 제시한다.

2. 관련연구

본 장에서는 본 논문의 기반이 되는 데이터 스트림과 자원 공유를 이용한 연속 집계 처리 기법인 BINT와 LINT에 대해 기술한다.

2.1 데이터 스트림 시스템

데이터 스트림은 센서들로부터 연속적으로 발생하고 빠르게 유입되는 대용량의 데이터이다. 그래서 새로운 형태의 시스템인 데이터 스트림 매니지먼트 시스템이 등장하였다.[2, 3]

데이터 스트림 매니지먼트 시스템은 연속질의를 시스템에 등록하고 입력되는 스트림 중에서 조건에 맞는 데이터만을 처리한다. 연속질의는 입력되는 모든 스트림 데이터를 한번에 처리 할 수 없기 때문에 슬라이딩 윈도우 연산을 포함하고 있다. 슬라이딩 윈도우는 질의 처리를 위해 버퍼를 이동하면서 데이터의 일부를 저장을 한다. 슬라이딩 윈도우의 종류는 튜플을 단위로 이동하는 튜플기반 슬라이딩 윈도우와 시간의 범위로 이동하는 시간기반 슬라이딩 윈도우가 있다.[1, 7] 그러나 슬라이딩 윈도우 연산은 집계 연산을 할 때 많은 연속질의가 있으면 윈도우가 중첩되어 계산하는 경우가 있다. 이와 같은 문제점 해결을 위해 자원 공유 연속 집계 처리 기법이 제안되었다.

2.2 스트림 시스템 집계 질의

스트림 환경에서 다중 질의처리를 위한 기존의 자원 공유 연속 집계 처리 기법으로 B-INT(Base-Interval) 알고리즘과 L-INT(Landmark-Interval) 알고리즘이 있다.[6, 7, 10]

B-INT는 삽입, 갱신, Lookup 단계가 있다. 삽입 단계에서는 최하위 레벨에 튜플들을 저장해 놓는다. 갱신 단계 에서는 최하위 레벨에 두 튜플의 집계 연산을 하여 상위계층으로 연산결과를 전달한다. 이 과정을 반복적으로 수행하여 최상위 계층까지 집계정보를 구성한다. 마지막 Lookup 단계에서는 범위 질의를 만족하는 값들을 찾아 질의를 처리한다. B-INT는 계층별로 집계 정보를 저장하기 때문에 메모리 유지 비용이 크다는 단점이 있다.

다음으로 L-INT 역시 3단계로 실행이 된다. 그러나 계층구조를 B-INT와 다르게 대칭형으로 집계결과를 저장하기 때문에 두 개의 영역만 검색하면 질의처리가 가능하다. L-INT는 대칭인 두 개의 계층구조를 사용하기 때문에 공간사용 비용이 B-INT보다 크다.

위의 두 기법은 공간 DSMS에서 고려되지 않아 공간 정보를 가진 환경에서는 적합하지 않다. 따라서 제안 기법에서는 공간 DSMS에서 R-tree 이용하여 연속질의에 포함된 공간영역 데이터를 색인으로 구축하여 질의의 단말 노드에 집계 값을 유지하는 기법을 제안한다.

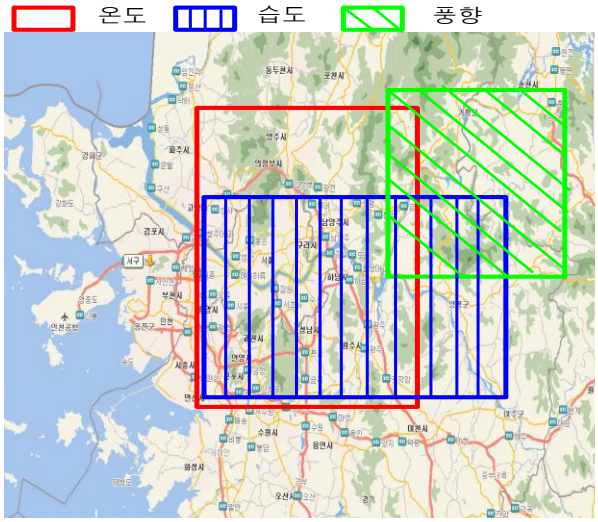
3. R-tree 기반의 공간 연속질의 처리를 위한 집계기법

본 장에서는 공간 영역 질의 처리를 위한 R-트리 기반의 집계 방법을 설명한다. 제안 기법은 DSMS에 등록된 공간 연속질의의 영역을 R-tree에 삽입하여 인덱스를 구축하여, 이를 이용하여 해당 스트림데이터에 대한 집계를 한다. 각 질의의 집계 값은 단말노드에 저장되어 있다. 본 장 3.1절에서는 R-트리 기반 집계 기법을 위한 구조를 설명하고, 3.2절에서는 실제로 데이터 스트림에 대한 집계 질의 처리 과정을 보여준다.

3.1 공간 스트림 데이터의 집계를 위한 R-tree구조

공간 DSMS에서는 다양한 종류의 GeoSensor로부터 위치 값을 포함하는 스트림데이터를 입력 받아 연속질의를 처리한다. 제안 기법에서는 등록되는 연속질의에서 해당 공간영역을 R-tree에 삽입하여 색인을 구축하며, 이는 공간 스트림데이터의 종류에 따라 각기 다른 R-tree로 구축된다. 예를 들어

"1시간마다 특정 지역의 평균 온도 값을 구하여라."라는 연속질의가 등록되면, 온도에 대한 R-tree를 생성하고 해당 질의의 영역에 해당하는 rectangle, polygon과 같은 공간 데이터를 삽입한다. 또한 풍속, 습도와 같이 온도가 아닌 다른 종류의 스트림데이터를 요구하는 연속질의가 등록되면 각각의 R-tree를 생성하고 위와 같은 방법으로 해당 공간 데이터를 삽입하여 색인을 구축한다. <그림 1>은 DSMS에서 온도, 습도, 풍향에 대한 공간영역질의 처리를 위한 R-tree 색인 구축 예이다.

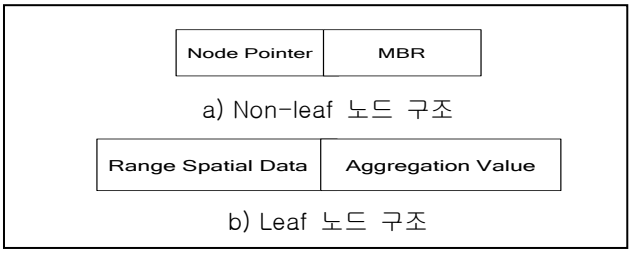


<그림 1> DSMS에서 온도, 습도, 풍향에 대한 공간영역질의 처리를 위한 R-tree 색인 구축 예

위 그림에서 각각의 rectangle은 공간 스트림데이터의 종류에 따라 구축된 R-tree의 루트 노드를 나타낸다.

3.2 R-tree 기반의 공간 연속질의처리를 위한 자료구조

본 장에서는 공간 스트림데이터의 종류에 따라 구축된 R-tree에서 하나의 R-tree에 대한 자료 구조를 설명한다. 공간영역을 포함하는 연속질의를 위한 R-tree는 질의 영역을 나타내는 rectangle, polygon과 같은 공간 데이터를 삽입하여 구축하며, 삽입 알고리즘은 기존의 R-tree와 같다. 이때 non-leaf 노드는 영역 공간 데이터를 포함하는 MBR(Minimized Boundary Rectangle)을 가지며, leaf 노드는 한 개의 영역 공간 데이터와 집계 값을 갖는다. <그림 2>는 공간영역의 집계 값 저장을 위한 R-tree의 자료구조이다.

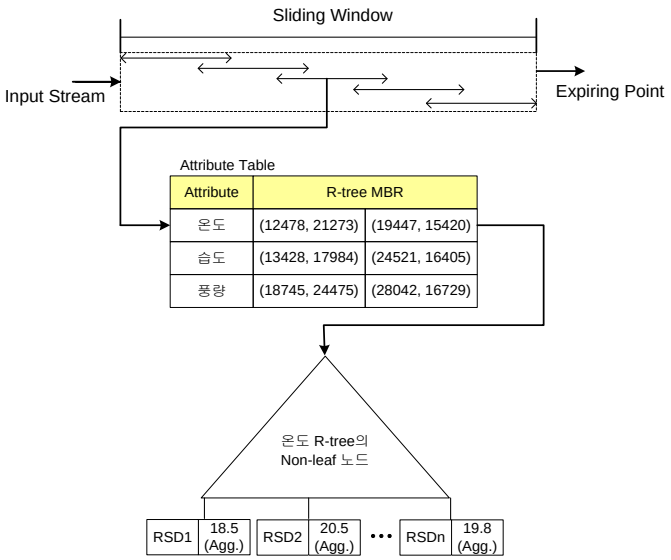


<그림 2> 공간영역의 집계 값을 위한 R-tree 노드 구조

Non-leaf 노드에서 Node Pointer는 자식 노드를 참조하기 위한 포인트 이며, MBR은 하위 노드의 영역을 포함하는 것으로 R-tree와 같다. Leaf 노드는 하나의 연속질의의 영역 공간 데이터를 Range Spatial Data에 저장하며, 질의에서 요청한 집계 값을 해당 주기에 따라 갱신하여 Aggregation Value에 저장한다. 집계 연산은 MAX, MIX, AVG, SUM을 고려한다.

3.3 공간 범위 질의 처리 과정

본 절에서는 공간 범위 질의 처리 과정을 기술한다. 데이터가 연속적이고 끊임없이 들어오는 데이터 스트림 환경에서 속성 테이블은 데이터 셋을 처음으로 필터링 하는 역할을 한다. <그림 3>은 R-tree 기반의 집계 값을 이용한 공간 범위 질의 처리 과정이다.



<그림 3> R-tree 기반의 집계 값을 이용한 공간 범위 질의 처리 과정

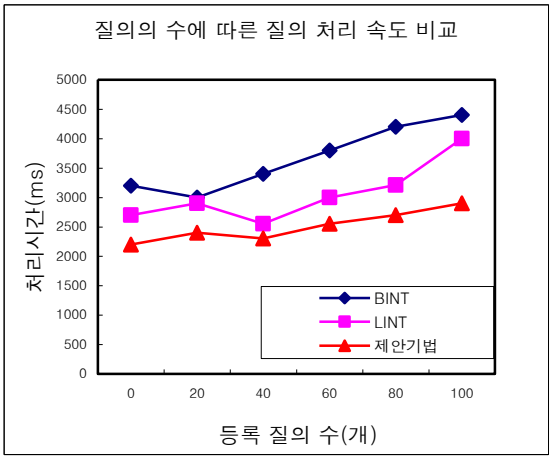
<그림 3>의 예에서 현재 들어오는 데이터 셋에 대해 온도, 습도, 풍향 이외의 값이 들어온다면 데이터 셋을 통과시키고 디스크에 저장한다. 또한 속성의 MBR 밖에 있는 센서에서 들어온 정보를 필터링 하는 역할을 한다.

속성테이블로 필터링 된 데이터 set을 2차 필터링을 하게 된다. R-트리 기반 질의 인덱스에서 각 질의의 MBR 밖에 있는 데이터들은 디스크에 저장된다. 그러나 질의 범위 안쪽에 있는 데이터일 경우에는 각 질의가 원하는 집계 연산을 하여 집계 값을 갱신하게 된다.

4. 성능평가

성능평가는 공간 정보를 포함하는 스트림 환경에서 질의 인덱스를 하지 않았을 때와 제안기법을 사용 할 때를 서로 비교 분석하였다. 모든 실험은 2GB 메모리의 인텔 펜티엄4 2.6GHz 프로세서와 윈도우XP 프로페셔널 운영체제 하에서 수행되었으며 알고리즘 구현은 Microsoft Visual Studio 2005에서 C언어로 구현하였다.

실험을 위해 초당 1만개의 튜플을 20초 동안 입력하고 질의의 수를 20개씩 증가 시켰다.



<그림 4> 등록 질의 개수에 따른 질의 처리시간 성능평가 결과

<그림 4>는 등록되는 질의의 개수에 따른 질의 처리 속도의 결과이다. BINT의 경우 처리시간이 가장 늦었고 LINT는 40개의 질의 수까지 BINT보다 좋은 성능을 보였지만 질의 수가 많아지면서 BINT와 비슷한 성능을 내기 시작했다. 그러나 R-tree 기반 집계 기법은 서서히 증가하는 것을 알 수 있다.

5. 결론 및 향후 연구

본 논문은 GeoSensor로부터 입력되는 데이터 스트림의 집계 값을 효율적으로 처리하기 위한 기법을 제안하였다. 제안 기법은 위치 값을 갖는 GeoSensing 데이터의 특성을 고려하여 공간영역 및 데이터의 중복이 발생함에 따라 연속질의처리 비용이 증가되는 문제를 해결하였다. 본 논문의 제안 기법에서는 R-tree를 이용하여 공간 범위 질의를 인덱싱 하고, 질의에 따른 버퍼를 두어 집계 값을 갱신하는 방법으로 공간영역 기반의 연속질의 처리 성능을 향상하였으며, 집계 데이터 만을 버퍼에 관리 함으로써 저장 비용을 줄일 수 있었다. 그러나 새로운 질의가 들어올 때마다 질의 인덱스를 갱신하는 비용과 질의가 종료되어 해당 공간 영역데이터가 R-tree에서 삭제될 경우 R-tree를 재구성하는 비용이 요구되는 단점이 있다. 향후 연구에서는 동일 속성 R-tree에서의 중첩된 영역에 대한 고려와 R-tree 색인의 재구성 비용을 줄이는 기법에 대한 연구가 필요하다.

6. 참고문헌

[1] J. Gehrke, F. Korn, and D. Srivastava. "On computing correlated aggregates over continual data streams," In Proc. of the ACM SIGMOD, pp. 13-24, 2001.

[2] B. Babcock, S. Babu, M. Datar, R. Motwani, and J. Widom., "Models and Issues in Data Stream Systems," Invited paper in Proc of PODS, 2002.

[3] D. Abadi, J. Carney, D. Centintemel, U. Cherniack, M. Convey, C. Lee, S. Stonebraker, M. Tatbul, and N. Zdonik, "Aurora: A New Model and Architecture for Data Stream Management," VLDB Journal, pp. 120-139, 2003

[4] A. Arasu, S. Babu, and J. Widom, "The CQL Continuous query Language : Semantic Foundations and Query Execution," Stanford University Technical Report, 2003.

[5] R. Motwani, J. Widom, A. Arasu, B. Babcock, S. Babu, M. Datar, G. Manku, and C. Olston, J. Rosenstein, and R. Varma, "Query Processing, Resource Management, and Approximation in a Data Stream Management System," In Proc of CIDR, 2003.

[6] A. Arasu, and J. Widom, "Resource Sharing in Continuous Sliding-Window Aggregates," In Proc. of the VLDB, pp.336-347, 2004.

[7] J. Li, and D. Maier, "No Pane, No Gain : Efficient Evaluation of Sliding-Window Aggregates over Data Stream," SIGMOD Record, pp.39-44, 2005.

[8] R. Zhang, and N. Koudas, "Multiple Aggregations Over Data Stream," In Proc. of the ACM SIGMOD, pp.299-310, 2005.

[9] Guttman, A., "R-tree: A dynamic index structure for spatial searching," Proc. of Intl. Conf. on Management of Data, ACM SIGMOD, 1984.

[10] J. Li, and D. Maier, "Semantics and Evaluation Techniques for Window Aggregates in Data Stream," SIGMOD Record, pp.39-44, 2005.