韓國컴퓨터情報學會 論文誌
第14卷 第2號, 2009. 2.

2009-14-2-3-1

범위 질의 인덱싱을 이용한 스트림 데이터의 다중 질의처리 기법

이 동 언*, 이 윤 석**

A Multi-dimensional Query Processing Scheme for Stream Data using Range Query Indexing

Dong-Un Lee*, Yunseok Rhee**

요 약

스트림 서비스 환경에서는 지속적으로 입력되는 막대한 양의 데이터에 대해 원하는 조건을 탐색하는 실시간 질의처리가 요구된다. 기존의 R-tee 기반 질의처리 기술은 각 이벤트에 대해 트리 전체에 대해 동일한 탐색과정을 반복해야 하므로 이를 효율적으로 감당할 수 없었다. 한편 센서 측정값을 비롯한 대부분의 스트림 데이터는 매우 높은 지역성을 가지며 이를 활용하여 탐색 공간을 크게 줄일 수 있다. 따라서 본 연구에서는 스트림 데이터의 지역성을 활용하여 스트림 환경에 적합한 질의처리 기법을 제안하였다. 또한 이 프레임웍을 활용하여 스트림 환경에서 어플리케이션이 요구하는 다양한 질의처리 서비스를 개발할 수 있을 것으로 기대된다. 본 연구에서 구현한 프로토타입 시스템을 스트림 환경에 적용해 얻은 실험 결과를 통해, 스트림 환경에서 기존 질의처리 기법보다 더 적합하고 효율이 크게 개선됨을 확인할 수 있었다.

Abstract

Stream service environment demands real-time query processing for voluminous data which are ceaselessly delivered from tremendous sources. Typical R-tree based query processing technologies cannot efficiently handle such situations, whichㅁㅁ require repetitive and inefficient exploration from the tree root on every data event. However, many stream data including sensor readings show high locality, which we exploit to reduce the search space of queries to explore. In this paper, we propose a query processing scheme exploiting the locality of stream data. From the simulation, we conclude that the proposed scheme performs much better than the traditional ones in terms of scalability and exploration efficiency.

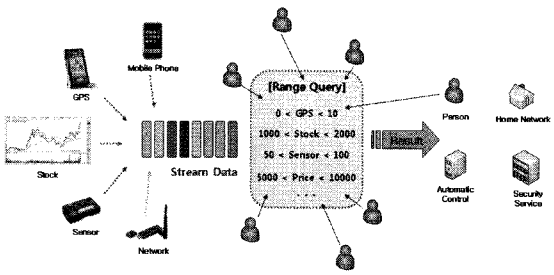
▸ Keyword : 스트림 데이터(stream data), 데이터 지역성(data locality), 범위 질의 인덱스(range query index), 다차원 질의(multi-dimensional query)

• 제1저자 : 이동언 교신저자 : 이윤석
• 투고일 : 2008. 12. 1, 심사일 : 2008. 12. 7, 게재확정일 : 2009. 1. 20.
* 한국외국어대학교 전자정보공학부 석사 ** 한국외국어대학교 전자정보공학부 교수
※ 이 논문은 2008학년도 한국외국어대학교 학술연구비 지원에 의하여 이루어진 것임.

Ⅰ. 서 론

유비쿼터스 컴퓨팅의 실현과 함께 센서네트워크, 모바일 네트워크 등의 다양한 매체를 통해 막대한 정보들이 유통되고 이를 위한 새로운 서비스가 출현하고 있다 [1,2,3,4]. 특히 [그림 1]과 같은 스트림 서비스 환경은 각종 데이터 단말들로부터 막대한 데이터들이 연속적인 스트림 형태로 쏟아져 들어오고, 많은 사용자들이 자신들이 원하는 데이터 또는 이벤트를 다양한 질의(query) 형태로 등록, 이를 실시간으로 제공받는 환경을 일컫는다 [5,6,7]. 이와 같은 환경에서는 수많은 사용자의 질의를 효과적으로 저장하고, 특정한 데이터(또는 이벤트)의 입력에 대해 이를 만족하는 질의를 효과적으로 찾아내어 해당 사용자에게 그 결과를 신속하게 제공하는 대규모 실시간 모니터링(massive realtime monitoring) 시스템의 개발이 필요하다.

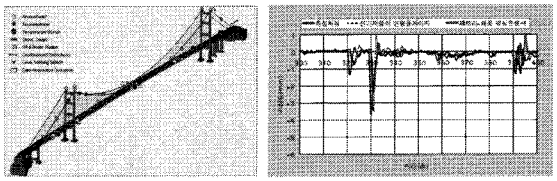
기존의 연구들은 대개 데이터베이스를 기반으로 데이터를 저장하고, 이를 사용자의 질의에 대해 효과적으로 자료를 검색 제공하는 형태로 이뤄졌으나[1,3,8], 주기적으로 발생하는 센서값 등을 DB에 저장하는 자체가 매우 무모하며, 대분의 데이터가 시간이 지나면 소용없게 되는 일과성 데이터인 점에서 DB활용의 필요성은 매우 낮다. 뿐만아니라 이와 같은 일과성 데이터가 엄청난 규모로 발생하는 상황에서 이를 저장 후 검색 방식으로 지원하는 것은 시스템의 응답성과 확장성을 떨어뜨리는 요인이기도 하다.



[그림 1] 스트림 서비스 환경
Fig 1. Stream Service Environment

스트림 서비스에 관한 연구는 현재 센서네트워크, 주식거래, 인터넷 옥션 등의 분야에서 활발히 행해지고 있다. 이 가운데 Bridge Health Monitoring 시스템[9,4,10]은 대표적인 적용 사례에 속한다. 기존에는 교량을 유지, 보수하기 위해 주기적으로 사람이 직접 안전도를 측정하고 보수하는 작업을 했으나, 사람의 이동 반경과 감각 정보에 한계가 있어 전체 교량을 세밀하게 점검하기가 어려웠고 작업 환경 또한 매우 위험했다. 하지만 센서(온도, 습도, 진동, 장력 등)의 종류가 많아지고, 이들 센서 간 네트워킹이 가능해지면서 이를 활용한 자동화된 교량 안전 시스템이 개발되었다. 즉, [그림 2(a)]에 보이는 바와 같이, 교량의 상태를 확인할 수 있는 중요한 지점에 많은 수의 센서들이 장착되고, 이들 센서들의 데이터들을 수집, 처리할 수 있게 되었다. 이를 통해 [그림 2(b)]처럼 수집된 센서 정보를 분석하여 교량의 상태를 쉽게 모니터링함으로 더욱 정밀하고 안전한 유지, 보수가 가능하게 되었다.또한 각 센서에 대해 다수의 사용자(또는 시스템)가 다양한 종류의 질의를 통해 교량의 상태정보를 제공받을 수 있게 되었다. 이처럼 수많은 질의들을 효율적으로 처리하여 스트림 형태로 입력되는 센서 정보를 원하는 사용자에게 신속하게 전달해주는 질의처리 시스템 설계는 매우 중요한 문제이다.

또 다른 대표적 적용 예는 주식 데이터에 대한 가격 변동을 모니터링하고 이를 사용자들에게 제공하는 경우이다. 매우 짧은 시간에 수많은 종류의 주가가 계속적으로 변동하고, 사용자는 자신이 매매를 원하는 주가를 기다리는 상황에서, 실시간으로 발생하는 수많은 주가 데이터는 스트림 데이터를 구성하며, 사용자가 원하는 주가 조건은 질의에 해당한다. 이와 같은 환경에서 주가 변동 정보를 매우 빠르게 사용자에게 알려주는 서비스 즉, 질의처리 결과를 매우 빠르게 알려주는 질의처리 시스템 설계는 필수적이라고 볼 수 있다.



(a) 교량 센서 설치 모형 (b) 센서 측정값 모니터링 화면
[그림 2] Bridge Health Monitoring 사례
Fig 2. Case of Bridge Health Monitoring

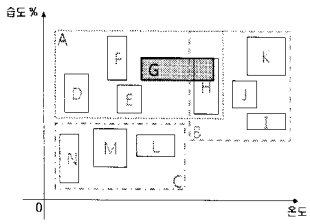
이처럼 다양하게 스트림 서비스가 활용되고 있으며, 이 서비스는 연속적인 스트림 데이터의 특성에 적합한 질의처리 시스템 설계를 통해 수많은 사용자들이 다양하게 질의를 등록함은 물론 빠르게 정보를 제공받는 것이 매우 중요한 목적이라고 할 수 있다. 따라서, 본 연구에서는 스트림 데이터의 높은 지역성(locality)를 활용하여 사용자의 다양한 질의 표현을

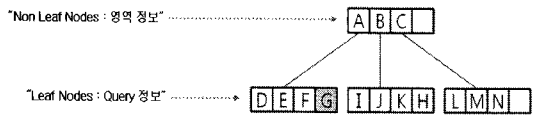
효과적으로 지원하는 다중 질의처리 기법을 제안하고자 한다.

II. 기존 질의처리 방법과 문제점

기본적인 스트림 처리를 위해서 B-트리를 활용한 1차원 질의처리 구조가 대표적으로 활용된다 [1,11,12]. 예를 들어 다수의 센서(온도, 습도, 조도, 등) 정보를 가지고 질의를 구성한다면 1차원 질의 처리 구조는 모든 정보(센서)의 질의를 한 곳에 관리하는 것이 아니라 온도질의, 습도질의, 조도질의 등과 같이 각각 나누어서 구성하는데 사용된다. B-트리 구조는 사용자가 등록한 질의가 트리의 각 단말노드에 저장되며, 질의범위의 시작값에 따라 정렬되어 구성된다. 따라서, 트리에 저장되어 있는 질의를 탐색하기 위해서는 최상위 노드부터 마지막 노드까지 트리의 높이만큼 탐색하여 효율적인 탐색 경로를 제공한다.

하지만 트리에 담겨진 질의 정보는 단일 데이터로 정해져 있어 범위 질의에 대한 질의는 등록 할 수 없다. 이에 1차원 질의처리 구조는 다양한 질의를 표현 할 수 없기에 스트림 서비스 환경에 적용하기 힘든 구조임을 알 수 있다.

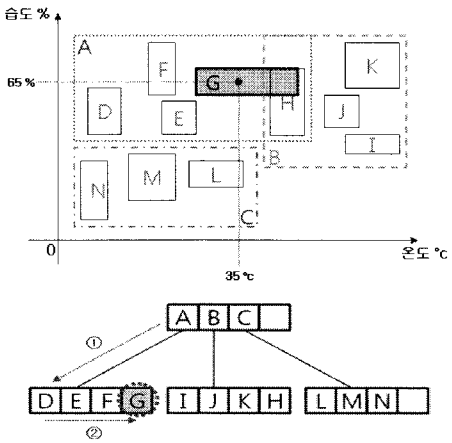




[그림 3] R-트리를 이용한 2차원 범위 질의 구성 예
Fig 3. Example of 2-D Range Query Construction

이를 개선한 방법으로 [그림 3]과 같이 R-트리를 사용, 2차원 범위질의를 표현하는 방식이 제안되었다 [1,2,3,10]. 다차원 질의처리 구조는 일차원 질의처리 구조에서 설명한 방법과는 다르게 모든 차원의 질의 정보를 한곳에 모두 저장 할 수 있는 구조이다. 즉 앞에서 예를 들었던 센서 정보를 가지고 질의를 구성한다면 온도, 습도, 조도 등의 모든 질의 정보를 표현한 질의처리 구조라고 할 수 있다.

[그림 3]의 2차원 질의처리 구조는 온도에 대한 범위 질의와 습도에 대한 범위 질의 조건이 동시에 표현된다. 이렇게 사용자가 등록한 범위 질의는 B-트리와 유사하게 단말노드에 저장되며 중간 노드들은 하위 노드의 범위를 포함하는 전체 영역 정보를 갖는다. R-트리 역시 질의의 범위의 시작값에 대해 정렬된 형태로 구성되기 때문에 최상위 노드부터 마지막 노드로 트리의 높이만큼 탐색하는 경로를 제공한다. 이처럼 R-트리 질의처리 구조는 범위 질의를 등록 할 수 있고 다차원 질의 처리가 가능하지만, 차원이 증가할 수록 각 노드 정보와 이로 인해 증가하는 하위 노드의 수가 기하급수적으로 증가하므로 3차원 이상의 다차원 구조에 적합하지 않다 [10].



[그림 4] R-트리에서의 탐색 과정
Fig 4. Traversal Process in the R-트리

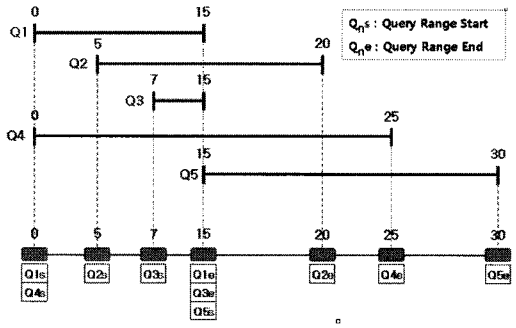
[그림 4]와 같은 2차원 질의처리 구조에서 해당 질의를 찾는데 걸리는 과정은 R-트리의 확장성에 대한 한계를 잘 설명한다. 예를 들어, 입력된 데이터가 온도 35 °C, 습도 55 %라고 할 때, 입력된 데이터를 만족하는 질의를 찾기 위해 트리의 루트 노드부터 포함 영역 정보를 확인하며 해당 질의를 탐색하게 된다. 탐색 과정을 보면 입력 데이터는 최상위 노드 A, B, C를 확인하여 A 영역에 만족 한 것을 알고 첫 번째로 A 노드에 방문한다. 그리고 A 노드가 가리키는 영역으로 이동한다. 이동한 영역에 있는 D, E, F, G 노드를 확인하고 만족하는 G 노드를 찾은 다음 더 이상 가리키고 있는 경로가 없다는 사실을 판단 한 뒤 모든 탐색을 종료한다. 결국 입력한 데이터가 질의 G를 만족하고 있다는 사실을 알려주게 된다.

한편, 기존 질의처리 방법에서는 입력된 데이터가 중복된 질의 영역에 있다면 트리의 높이만큼만 탐색하는 것이 아니라

만족되는 질의가 끝날 때까지 트리의 노드를 계속 탐색해야 한다. 스트림 서비스 환경에서는 수많은 데이터와 매우 다양한 범위 질의가 등록되기 때문에 대부분 질의가 중복된다. 결국 질의를 모두 찾기 위해서는 트리의 노드를 계속 방문해야 하기 때문에 처리 시간이 늦어지는 단점을 보이게 된다. 또한 R-트리는 언제나 각 차원(온도, 습도)에 대한 데이터가 모두 입력되어야만 탐색을 수행하는 단점이 있다. 그리고 데이터가 입력 될 때마다 매번 최상위 노드부터 다시 탐색해야하기 때문에 스트림 환경에서 효과적인 탐색 결과를 보장하기 어렵다는 사실을 알 수 있다.

III. 제안하는 다중질의 처리구조

본 연구에서는 보다 효과적인 스트림 기반 질의처리를 지원하기 위해 [그림 5]에 보이는 사용자 범위 질의 정보를 담고 있는 질의 인덱스를 설계하고, 이를 범위 질의 인덱스(Range Query Index)라고 부른다. 범위 질의 인덱스는 기존 방법과 유사하게 범위 질의를 담고 있으나, 그림에 보이는 것처럼 질의의 범위 데이터들을 정렬한 형태로 인덱스를 구성하고 있고, 그 인덱스에는 질의 정보의 범위 시작 정보와 종료 정보가 담긴 점이 다르다.

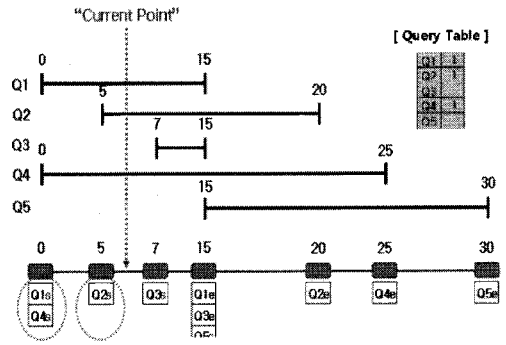


[그림 5] 범위 질의 인덱스(Range Query Index)의 예
Fig 5. Range Query Index Example

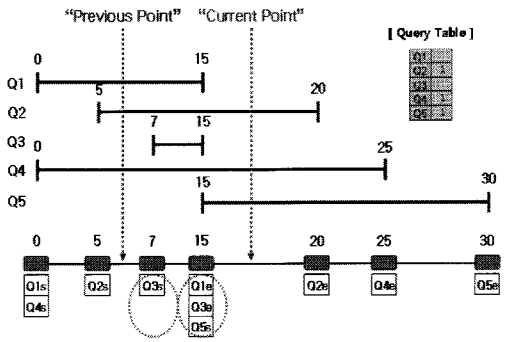
예를 들어 범위 질의 Q1(0 ≤ value ≤ 15)을 등록하면 범위 데이터인 0과 15 가 인덱스를 이루고, 인덱스 0에는 질의 범위의 시작지점이 등록되고, 인덱스 5에는 질의 범위의 종료 지점을 등록한다. 새로운 질의가 등록 될 때마다 동일한 과정을 거쳐 각 질의를 등록하는데, 예를 들어, 또 다른 범위 질의 Q2(0 ≤ value ≤ 30)와 같이 동일한 지점(즉 '0')를 포함하게 되는 질의가 등록하게 되면 그림과 같이 Q1에 대한 범위 질의 정보 다음에 등록해서 질의를 구성하면 된다. 결국 기존의 R-트리를 이용한 질의처리 방법(질의정보 + 영역정보)과는 다르게 범위 질의 정보만으로 질의처리 구조를 완성하게 된다.

3.1 1차원 범위 질의 인덱스

[그림 6]은 앞서 설명한 등록 방법으로 1차원 범위를 갖는 질의 5개의 질의 인덱스를 갖는 예제이다. 위 예제에서 특정 스트림 데이터의 입력으로 만족되는 질의(들)를 찾는 과정은 다음과 같다.



(a) 입력 데이터 6인 경우



(b) 입력 데이터 17인 경우

[그림 6] 입력 데이터 값에 따른 질의 탐색과정의 예
Fig 6. Query Search Process Example with Input

먼저 [그림 6(a)]에 보이는 것처럼, 데이터 6이 입력되면 범위 질의 인덱스를 차례로 탐색하게 되는데, 인덱스 0부터 인덱스 5까지 방문하게 되고, 인덱스 7은 입력된 6 보다 큰 인덱스이기 때문에 더 이상 방문하지 않는다. 인덱스 0을 방문할 때 Query 1과 Query 4가 만족(시작)되었다는 정보를 확인하고 Query Table에 해당 Query에 1로 비트 플래그를 설정(set)한다. 결국 인덱스 5까지 이동하면서 Query 1, 4, 2의 플래그가 순차적으로 설정되고, 그 정보는 데이터 6에 의

해 만족되는 질의가 Query 1, 4, 2 라는 결과를 알려준다. 결국 인덱스 단계를 이동하면서 Query 1, 4 만족 → Query 2 이라는 사실 또한 알려주게 된다.

이후 후속 스트림 데이터가 입력되면 기존 질의 처리 방법과는 다르게 이전 질의 정보를 담고 있는 Query Table을 가지고 있기 때문에 인덱스 처음부터 찾는 것이 아니라 전에 만족되었던 인덱스 위치부터 탐색을 시작하면 된다. 예를 들어 [그림 6(b)]처럼 다음 데이터로 17이 입력되면 이전에 인덱스 5까지 방문하였기 때문에 이 지점부터 위와 같은 방법으로 탐색을 시작한다. 결국 7 → 15로 인덱스를 차례로 방문하게 된다. 인덱스 7에서는 Query 3의 플래그를 설정하면서 Query 3이 시작되고 있다는 사실을 알려준다. 다음 인덱스인 15에서는 Query 1, 3이 끝나는 정보와 Query 5가 시작되었다는 정보를 가지고 Query Table 에 해당 Query 1, 3을 0 으로 초기화(reset)하고 Query 5의 플래그를 1로 설정하면서 Query 1, 3 에서 벗어나고 Query 5에 진입했다는 사실을 알려준다. 결국 입력 데이터 17은 앞에서와 같이 만족되는 질의가 Query 2, 4, 5 라는 결과를 알려주며 그 사이에 Query 1, 3에서 벗어났다는 사실 또한 알려준다.

[표 1] 질의 결과값과 그 의미
Table 1. Query Results and Meanings

Query Result Types	Query Status
Stay	만족하는 질의에 머물러있다.
Step In	만족하는 질의에 들어왔다.
Step Out	만족하는 질의에서 벗어났다.

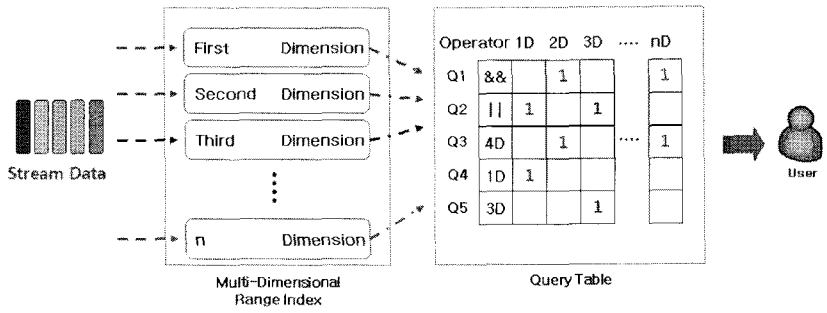
[그림 6]과 [표 1]은 Query Table에서 위에서 설명한 탐색 과정 결과 각각의 질의가 1 과 0 으로 Set, Reset 될 때 질의의 변화에 대한 상태를 보여주며, 이와 같이 기존 질의처리 방법은 만족하는 질의만 알려주는데 반해 범위 질의 인덱스 탐색 방법은 질의의 변화를 모두 알 수 있는 장점을 가져다준다.

3.2 다차원 범위 질의 인덱스

다차원 질의처리를 구성하기 위해 기존 처리 방법에서는 모든 차원의 범위 정보를 하나의 질의 처리 시스템에 구성하여 매우 높은 복잡도를 갖는다. 하지만 본 연구에서 제안하는 범위 질의 인덱스 기법은 [그림 7]과 같이 1차원 범위 질의 인덱스를 각 차원별로 각각 구성하여 다차원 질의 처리를 구성하였다. 그리고 질의 정보를 알려주는 Query Table 또한 각 차원별로 제공하고, 각 차원에 대한 표현 방법을 알려주는 Operator를 두어서 Query 표현(온도 && 습도, 온도 || 습도, 온도, 습도 등)을 다양하게 등록 할 수 있도록 하였다. 이와 같이 각 차원 마다 독립적으로 질의 정보를 구성하기 때문에 질의의 확장을 편리하게 할 수 있는 장점이 있다.

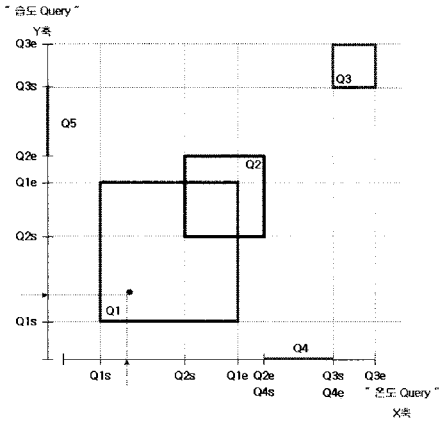
한편 [그림 8]은 다차원 범위 질의 인덱스에서 2차원인 경우를 보이고 있는데, 각각의 차원이 독립적으로 구성되어 있기 때문에 Q1처럼 두 개의 범위 조건이 모두 만족하는 경우와 Q5처럼 한 차원의 조건만 만족하는 범위 질의를 모두 등록 할 수 있다. 기존 질의처리 기법에서는 Q5를 등록하기 위해서는 모든 차원이 전부 등록해야 하는 단점이 있었지만, 다차원 범위 질의 인덱스에서는 각각의 차원이 독립적으로 표현되기 때문에 한 차원만 등록하면 된다. 결국 기존 질의처리 기법에서 질의 표현을 위한 불필요한 용량을 줄일 수 있게 된다.

[그림 9]는 2차원 범위 질의 인덱스에서 탐색 과정을 나타내고 있다. 1차원과 같은 방법으로 각 차원마다 탐색을 진행하고 각각의 질의 상황을 Query Table 등록함으로서 질의의 상황 변화를 알려줄 수 있다. 또한 Operator 등록으로 사용자 질의의 표현을 다양하게 지정 할 수 있다. 예를 들어

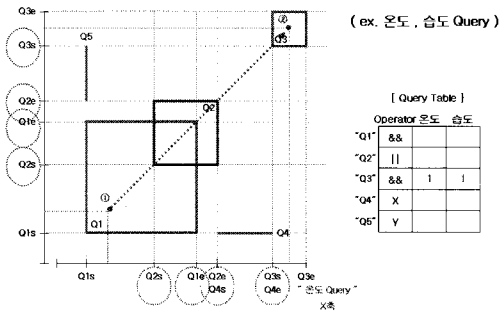


[그림 7] 다차원 범위 질의 인덱스 구성과 동작 예
Fig 7. Multi-dimensional Query Index and Operation Example

Query 2 는 || 를 등록함으로서 Query Table에서 Query 2에 대한 온도, 습도 둘 중에 하나만 Set 되면 곧바로 알려주면 되는 것이다. 그리고 1차원과 동일하게 Query Table이 인덱스를 단계적으로 이동하면서 변화되기 때문에 [그림 9]에 보이는 바와 같이 첫 번째 데이터가 입력되고 두 번째 데이터가 입력되었을 때 첫 번째 데이터가 만족하는 Query 1과 두 번째 데이터가 만족하는 Query 3만 알려주는 것이 아니라 Query 2가 지나왔던 사실 또한 알려줄 수 있다. 다차원 범위 질의 인덱스도 역시 기존 질의 처리 방법에서 해결되지 못한 입력 데이터 변화에 대한 질의 변화를 알 수 있어 스트림 서비스 환경에 보다 적합하다는 것을 알 수 있다.



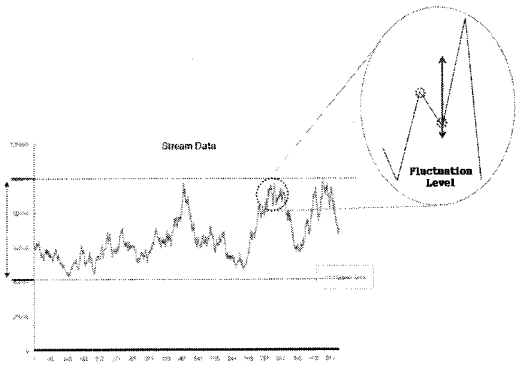
[그림 8] 2차원 범위 질의 인덱스 구성의 예
Fig 8. 2-D Query Index Construction Example



[그림 9] 다차원 범위 질의 인덱스의 탐색 과정 (2차원 예)
Fig 9. Search Example in Multi-dimensional Range Query Processing (2-D case)

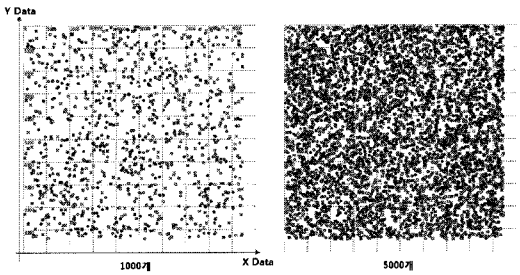
IV. 성능 평가

4.1 모의실험 환경



[그림 10] Fluctuation을 갖는 스트림 데이터 예
Fig 10. A Stream Example with Fluctuation

본 연구의 모의실험에 사용할 스트림 데이터를 생성하기 위한 특징은 다음과 같다. 실제로 지역성을 가지는 스트림 데이터의 특징을 나타내기 위해 Fluctuation Level을 설정하여 스트림 데이터를 생성하였다. Fluctuation Level은 현재 입력된 데이터 다음에 입력될 데이터가 나타나게 되는 범위로써 실제 스트림 데이터의 특징을 표현하기 위해 설정하였다. [그림 10]과 같이 Fluctuation Level의 변화를 두어 지역성을 가지는 경우와 아닌 경우 제안한 질의처리 시스템이 스트림 환경에 얼마나 적합한지를 측정하기 위해 Fluctuation Level (0.1 ~ 1.0%)를 두어 Uniform Random Generation 하였다. 그리고 Stream Data 는 10,000개로 동일하게 실험 하였다.



[그림 11] 사용자 범위질의의 생성 예
Fig 11. A Population of User Range Queries (Left: 1000, Right: 5000)

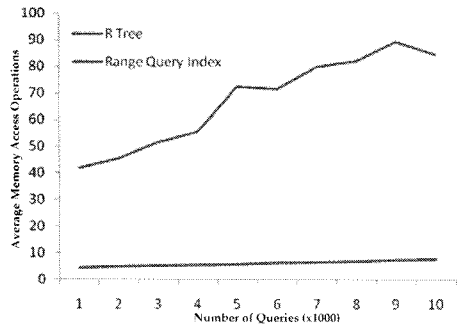
사용자 질의는 [그림 11]에 보이는 바와 같이 2차원 질의를 기반으로 World Space 0 ~ 10,000 범위 안에서 Uniform Random Generation (Generation Tool : Spatial Data Generator (c), University of Pireaus) 프로그램을 사용하여 무작위로 생성하였다. 질의를 1,000 ~ 10,000개 까지 생성하면서 실제 스트림 환경에서 질의의 수가 질의처리에 미치는 영향에 대해 실험하였다.

4.2 실험 및 성능 분석

범위 질의 인덱스에 대한 질의처리 성능 평가를 위해 질의의 수, 질의 범위, 스트림 데이터의 변화 범위를 각각 변화시키면서 실험을 진행하였다.

가. 질의의 수에 따른 성능

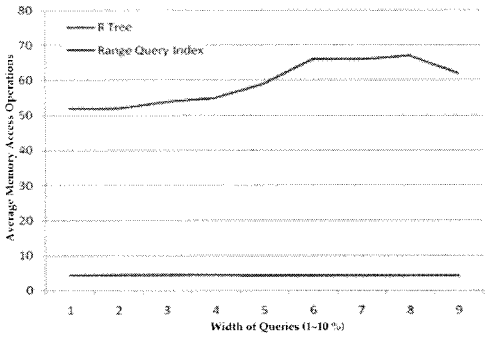
[그림 12]은 스트림 환경에서 질의의 개수가 질의처리 성능에 미치는 영향을 보여주는 그래프이다. 그래프에서 보듯이 R-트리의 경우 질의의 개수가 많아질수록 탐색하는 영역이 많아지게 되고 중복되는 영역도 늘어나기 때문에 질의 탐색 횟수는 계속 늘어나게 된다. 하지만 제안한 범위 질의 인덱스 방법은 질의의 개수에 따른 영향 보다는 스트림 데이터의 변동률 즉 지역성에 따라 인덱스를 탐색하는 횟수가 결정되기 때문에 지역성을 가진 스트림 데이터가 입력되면 거의 변동 없이 질의를 찾게 된다.



[그림 12] 질의 수에 따른 성능 변화
Fig 12. Effect Of the Number of Queries
(Data Fluctuation = 0.1%, Query Width = 1%)

나. 질의범위의 변화에 따른 성능

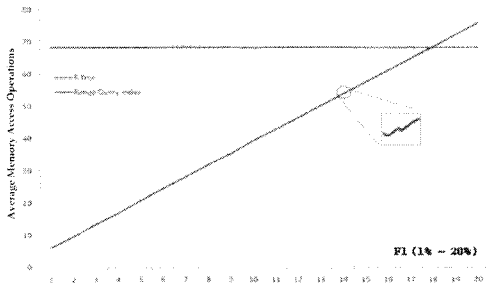
[그림 13]는 스트림 환경에서 질의의 범위가 질의처리 성능에 미치는 영향을 나타낸 그래프이다. 이 그래프 역시 지역성을 갖는 스트림 데이터이기 때문에 범위 질의 인덱스는 거의 변동 없이 일정하게 질의를 찾게 된다. 하지만 R-트리는 질의의 범위가 증가함에 따라 질의가 중복되는 경우가 높아지기 때문에 질의 탐색을 할 경우 탐색이 증가하게 된다.



[그림 13] 질의범위에 따른 성능 변화
Fig 13. Effect of The Width of Queries
(Fluctuation = 0.1%, Number of Queries = 1000)

다. 스트림의 변화폭에 따른 성능

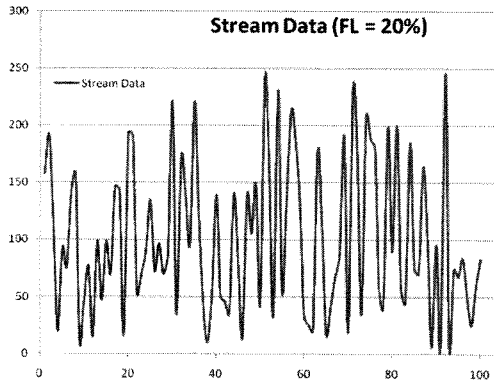
[그림 14]는 스트림 환경에서 스트림 데이터의 지역성에 대한 변화가 질의 처리 성능에 미치는 영향을 나타낸 그래프이다. R-트리의 경우 질의 중복에 대해서는 탐색 횟수가 많아지지만 중복이 없는 경우에는 트리의 높이만큼만 탐색하면 되기 때문에 중복이 적을수록 탐색 횟수는 줄어든다. 이처럼 R-트리에서는 스트림 데이터의 지역성에 대한 변화는 탐색 횟수에 영향을 주지는 않는다. 하지만 범위 질의 인덱스의 경우 스트림 데이터의 지역성이 크면 클수록 탐색하는 인덱스는 점점 많아지기 때문에 탐색 횟수는 계속 증가하게 된다.



[그림 14] 데이터 변화정도에 따른 성능 변화
Fig 14. Effect of The Fluctuation Level
(Number of Queries = 1000, Query Width = 1%)

한편 [그림 15]과 같이 입력 데이터가 스트림 형태가 아닌 지역성이 매우 큰 환경에서는 제안한 범위 질의 인덱스는 일정한 탐색 횟수를 보장 받지 못한다. 결국 이러한 환경에서는 기존 질의처리 기법보다 탐색 횟수가 커지는 상황을 보여주게 된다.

실험 결과에서 보듯이 지역성이 적은 스트림 데이터가 입력되는 스트림 환경에서는 제안한 범위 질의 인덱스가 기존 질의처리 방법보다 더욱더 빠르고 안정적인 질의처리 결과를 얻는다는 것을 알 수 있다.본 실험에서는 앞서의 모의실험 환경 하에, 기존의 R-트리를 사용한 방법과 본 연구에서 제안한 방법의 성능을 비교하였다.



〔그림 15〕 스트림 데이터 표본
Fig 15. Samples of Stream Data
(Fluctuation Level : 20%)

Ⅵ. 결론

본 논문에서는 기존의 질의처리 방법에서 문제가 되었던 스트림 환경 기반의 입력 데이터에 대한 질의 탐색 성능과 다차원 질의 표현에 대한 성능 개선을 연구하고 제안하였다. 그리고 범위 질의 인덱스를 설계하여 스트림 서비스 환경에 효율적인 질의처리 시스템을 구성하였고, 이를 바탕으로 모의실험을 통해 기존 질의처리 방법과 성능을 비교한 결과, 스트림 환경에서는 범위 질의 인덱스 방법이 다양한 질의 표현이 가능하고, 효과적으로 질의를 탐색함을 알 수 있었다.

향후 연구 방향으로 본 논문에서 제안한 범위 질의 인덱스를 실제 스트림 서버 아키텍처를 구현하여 실제 사용자가 다중 질의를 등록하는 환경에서 성능 평가를 하는 것이다. 또한 실제 스트림 데이터를 이용하여 질의처리에 대한 성능 평가를 추가적으로 수행해야 한다.

참고문헌

〔1〕 Guttman, "R-Trees: A Dynamic Index Structure for Spatial Searching," Proc. ACM SIGMOD, pp. 47-57, June 1984.

〔2〕 Mladen Bestvina, "*R-Trees in topology, geometry, and group theory,*" Handbook of geometric topology R. J. Daverman and R. B. Sher (editors), January 18, 1999.

〔3〕 Timos Sellis, Nick Roussopoulos and Christos Faloutsos, "*The R+ Tree: A dynamic index for multi-dimensional objects,*" Department of Computer Science University of Maryland College Park, 1987.

〔4〕 Stefan Berchtold, Daniel A. Keim, Hans-Peter Kriegel, "The X-tree: An Index Structure for High-Dimensional Data," Institute for Computer Science, University of Munich, The 22ndLVDB Conference Mumbai (Bombay), India, 1996.

〔5〕 J. Lee, Y. Lee, S. Kang, S. Lee, H. Jin, B. Kim, J. Song, "BMQ-Index: Shared and Incremental Processing of Border Monitoring Queries over Data Streams," Korea Advanced Institute of Science and Technology, MDM'06, 2006.

〔6〕 The Medusa Distributed Stream-Processing System, Magdalena Balazinska, Hari Balakrishnan, Jon Salz and Mike Stonrbraker, MIT Computer Science and Artificial Intelligence Lab, *http://nms.lcs.mit.edu/projects/medusa/*

〔7〕 Kim, W., "*Completeness Criteria for Object-Relational Database Systems,*" UniSQL Inc., 1996.

〔8〕 Ron Avnur, "Eddies : Continuously Adaptive Query processing," Univ of Berkeley, IEEE Network. January/February 2004.

〔9〕 서 석호, "*고속 스트림 데이터에 대한 연속 질의 처리 아키텍처의 설계,*" 한국 외국어 대학교 전자정보공학과 석사학위논문, 2005년.

〔10〕 이 석호, "*파일 구조,*" 정익사, 2007년.

〔11〕 한국철도기술연구원 2005. 11. 10 기술 소개 , "*교량에 설치된 센서의 종류 및 위치,*" 2005년.

〔12〕 건설기술정보, "*계측 센서 별 측정값 비교,*" 2001년.

저 자 소 개

이 동 언 (Dong-Un Lee)

2006년 2월: 한국외국어대학교 전자정보공학부 학사

2008년 2월: 한국외국어대학교 전자정보공학부 석사

관심분야: 임베디드시스템, 센서네트워크

이 윤 석 (Yunseok Rhee)

1988년 2월: 서울대학교 계산통계학과 학사

1999년 2월: 한국과학기술원 전산학과 박사

1999년-현재 : 한국외국어대학교 교수

관심분야: 분산시스템, 임베디드컴퓨팅