클러스터링을 이용한 공간 색인 기법 비교 분석

신 윤숙, 김 민수, 이 기준
부산대학교 전자계산학과

A Comparative Study on Spatial Indexing Methods using Clustering

Youn-suk Shin, Min-Soo Kim and Ki-Joune Li
Department of Computer Science, Pusan National University
e-mail : {yssin, mskim, lik}@spatios.cs.pusan.ac.kr

요약

지리 정보 시스템은 지리적, 위상적 정보를 가진 데이타를 한 번에 대용량으로 처리한다는 특성을 가지고 있다. 이 때문에 많은 지리 정보 시스템에서는 자료 처리의 용이함과 보다 빠르게 정보를 검색하기 위해서 부가적은 색인 자료 구조와 공간 색인 기법을 지원하고 있다. 현재까지 많은 공간 색인 기법들이 제안되고 있으나, 절대적인 평가 기준을 가지고 각 기법간 성능을 상호 비교, 분석하는 연구들은 제대로 이루어 지고 있지 않고 있다. 본 연구는 지리 정보 시스템에서 처리 되어야 할 정보들은 각 검색 작업이 공간적 근접성에 기반하고 있다는 사실로 부터 출발한다. 즉, 공간적 근접성을 잘 만족하는 것이라면 우수한 성능을 보인다는 점을 예상 할 수 있기에, 공간적 근접성을 기준으로 하여 색인 기법간 성능을 비교, 분석하고자 한다. 따라서 본 논문은 이에 따라 공간적 근접성을 평가 기준으로 클러스터링 알고리즘을 제안하고, 이를 이용하여 성능을 비교, 분석하고 있다.

1. 도입

수십 Giga에서 수 Tera에 이르기 까지 방대한 양의 지리 정보를 다루어야 하는 지리 정보 시스템에서 공간 색인 기법은 정보 처리의 용이함과 빠른 검색을 위해서 반드시 필요하다.[1]이러한 공간 색인 기법의 필요성은, 지금 현재까지 많은 수의 기법들이 제안되어 왔다.[2] 그러나 제안된 대부분의 색인 기법들이 특정 데이타 특성과 분포 상태, 그리고 처리하고자 하는 질의의 종류에 의존하여 구현된 것들이고, 각기 상이한 방식으로 구현된 자료구조들을 가지고 있기 때문에, 비교를 위해 사용되는 테스트 데이타의 특성에 따라 각기 다른 비교 결과가 보이고 있다.[3] 구체적으로 서술하자면 사용되는 데이타가 점들로만 구성된 일차원적 정보만을 가지고 있을 경우 우수한 성능을 보이는 색인 기법과, 선, 면등 일차원 이상의 정보를 가지고 있는 데이타를 처리 했을 경우 우수한 성능을 보이는 기법들이 일치하지 않는다. 이러한 현상은 데이타의 분포 상태, 즉 데이타가 전체 면적이 균등하게 분포하는가 한 곳에 집중적으로 분포하는가에 따라서도 역시 동일한 현상을 보인다. 마찬가지로 비교를 위한 시스템의 공통적인 환경을 설정하는 데에도 많은 어려움이 있다. 따라서 각각의 기법들을 비교하기가 쉬지 않으며, 이러한 이유로 인하여 여기에 대한 연구가 제대로 이루어 지지 않고 있는 실정이다. 그러나 지리 정보 시스템에 대한 중요성과 관심이 고조되고 있으며, 적용 분야가 확대됨에 따라 각각의 특정 임무에 따라 시스템을 구성하기를 요구받는 시스템 설계자들이 설계할 시스템의 특성에 맞는 공간 색인 방법의 선택이 용이 할 수 있도록 다양한 조건하에서 각각의 기법들을 비교, 분석함으로서 각각의 기법들의 특성을 잘 이해하는 일은 매우 중요하다고 할 수 있다.

본 연구는 지리 정보들에 대한 검색 작업들이 일반적으로 정보의 공간적 근접성에 그 기반을 두고 있다는 사실을 근거로 하여 이를 기본 척도로 두고 각 기법들간의 성능을 비교하고자 한다.[2]

본 논문은 구성은 다음과 같다. 2장에서는 본 연구에서 비교 대상으로 두고 있는 공간 인덱싱 기법을 간략하게 소개하고, 3장에서 비교, 분석 작업에 이용하게 될 공간적 근접성 이용한 데이타 클러스터링의 방법과 알고리즘을 서술하고 있다. 4장에는 데이타 클러스터링을 이용한 공간 색인 기법들간의 비교 방법들을 설명하고, 5장에서 그에 따른 결과를 가지고 각 기법들을 비교 분석하고 있다. 6장에서 마지막으로 결론 및 향후 연구 방향에 대해서 알아 본다.

2. 공간 색인 기법들

시스템 내에 처리 해야 할 지리 정보에 대한 저장 및 검색 방법들은 시스템이 지원하는 색인 자료 구조에 의해 결정 된다. 따라서 색인 자료 구조와 색인 기법들은 전체 시스템의 성능을 결정하는데 많은 영향을 미치게 된다.

공간 인덱싱 기법들을 간단하게 선형 공간 색인 기법과 다차원 공간 색인 기법으로 분류 할 수 있다.[4] 선형 공간 색인 기법은 함수를 사용하여 다차원의 공간을 일차원의 점으로 맵핑하는 방법으로써, N-order Peano Curve와 Column Wise Scan 기법등이 있다. 이에 반하여 다차원 공간 기법은 다차원의 공간을 적당한 방법을 사용하여 몇 개의 작은 공간으로 나누는 방법을 사용하고 있는데 Gird File Method, R-tree, R+-tree등이 있다.

본 논문에서는 다차원 공간 색인 기법들 중 대표적인 공간

색인 기법인 다음 세 가지 기법에 대해서 서로간의 성능을 비교하고자 한다.

- R-tree
- R+-tree
- R*-tree

R-tree는 B'-tree를 다차원으로 확장한 방법으로써 각각의 각각의 공간 객체들을 최소 경계 사각형 (MBR:Minimal Bounding Rectangle)로 나타내고 있다.[5] 데이타 노드(leaf node)들은 최소 경계 사각형과 사각형내 포함되는 객체를 가리키고 있는 포인터들로 구성되어 있으며, B'-tree와 비슷한 방식의 균형 트리이다. 트리의 각 노드를 구성하는 데이타 노드들은 그들과 위치상 근접한 공간 객체들을 모아서 한 데이타 페이지 내에 저장되며 R-tree의 디렉토리 노드는 (rect, children)로 구성되는데, rect는 이 노드의 최소 경계 사각형을 가리키고, children은 이 페이지들을 참조하기 위한 포인터 뿐만 아니라 자식 노드의 최소 경계 사각형에 대한 정보도 가지고 있다. R-tree는 다차원의 속성을 가지는 객체들을 처리가 용이하며, 객체간 공간적 근접성을 보장하는 장점을 가지는 반면에 디렉토리 노드를 구성하는 최소 경계 사각형들이 서로 겹치짐 현상이 발생하는 단점이 있다.

R'-tree 는 R-tree 의 변종 중 하나로써, R-tree에서 최소 경계 사각형간 서로 겹쳐지는 현상을 제거함으로써 데이타 검색 속도를 향상시킨 기법이다.[6] R'-tree는 R-tree가 가지는 단점을 제거함으로써 보다 향상된 성능을 보이고 있으나, 겹치지 않으면서 근접성을 유지하는 최소경계사각형을 찾아 내기가 어렵다.

R*-tree는 R-tree와 동일한 방식의 색인 방법이다. 그러나 최소 경계 사각형의 면적과 그들간 겹치지는 면적을 최소화하고 가능한한 정사각형에 가까운 최소 경계 사각형을 구현함으로써 R-tree가 가지는 문제점을 최소화 하였다.[7] 따라서 R-tree보다 약간 더 많은 삽입 비용으로 구축하지만 더 나은 검색 결과를 제공한다.

3. 데이타 클러스터링

지리 정보 시스템에서는 처리되는 질의는 객체의 지리적 특성을 기반으로 한다. 일반적으로 데이타 베이스에서 처리되는 질의는 다음 두 가지 유형으로 나눌 수 있다. 1) 주어진 질의의 조건을 정확하게 만족하는 정보의 검색을 요구하는 질의(exact match queries)이고 2) 주어진 질의의 조건의 범위 내에 속하는 정보를 전부에 대한 검색을 요구하는 질의(range queries)이다. 예를 들면

1) " 주어진 선의 끝점과 만나는 모든 선들을 찾아라 "

2) " 소방서로 부터 5Km이내 위치한 모든 집들을 찾아라 "

이다. 두 가지 유형의 질의들을 분석하면 둘 다 객체간의 공간적 근접성에 대한 요구들을 내포하고 있음을 알 수 있다. 만일 공간 색인 기법에 이러한 공간적 근접성을 이용한다면 성능에 큰 영향을 미치는 디스크 액세스 횟수가 줄어 듦에 따라, 나은 성능 향상을 가져 올 수 있을 것이다. 이러한 사실로 부터 공간적 근접성을 색인 기법들간의 성능을 평가하는 하나의 척도로 사용 가능함을 알 수 있다.

데이타 클러스터링은 데이타 베이스에서 성능 향상을 위해서 많이 사용되는 방법이다. 일반적인 데이타 클러스터링 알고리즘은 질의 처리시 디스크에 페이지 단위로 쓰여 있는 디스크 액세스 수를 줄이기 위해 자주 쓰이는 방법이다.[8] 본 연구에서는 각 공간 색인 기법간의 비교 대상으로 사용하기 위해서 다음과 같은 데이타 클러스터링 알고리즘을 제안한다. 아래에서 제안하고 있는 데이타 클러스터링 방법은 데이타들을 서로 근접한 곳에 위치한 것들끼리 같은 페이지 내 묶고, 페이지 단위로 화일에 쓰기를 함으로써 최적의 공간적 근접성을 만족하는 데이타 화일을 생성하고 있다.

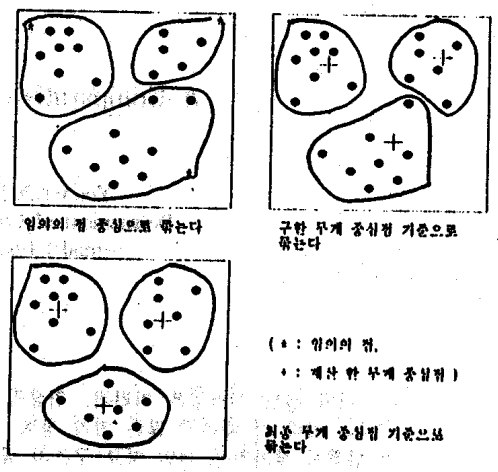

그림 1 데이타 클러스터링 과정

```
Algorithm Clustering( )
 input : point :  array of point objects
         num :  Number of points
         N :  the number of clusters
 output : cluster - array of clusters
 begin
  initialize the three initial Centers of Gravity;
    repeat
        for  each point object do
           Add point into the cluster with nearest
           Center of Gravity;
        Claclulate new center of gravity for each
        cluster;
    Until( There is No change of Center of Gravity)
 end

 Algorithm Make Cluster
 input : point :  array of points objects
         num :  the number of points
         N :  the number of cluster
 output : cluster - array of clusters
 begin
     N <- No

     For each center of gravity
        Compare number of datas with blocking factors;
        If cluster(point, num, N, clusters) there exits
        cluster with objects more than blocking factor;
       else
           Store cluster to file and exit;
       end if
   end for
 end
```

4 비교 분석 방법 구현

앞 장에서 구현하고자 하는 R-tree, R'-tree 그리고 R*-tree는 동일하게 클러스터링에 대한 색인 트리 구성이 완료된 후, 한 데이타 노드에 묶여진 데이타들을 페이지 단위로 디스크에 저장한다. 한 페이지내에는 서로 근접한 데이타들만을 포함하고 있기 때문에 일정 정도의 공간적 근접성을 보장하고 있다. 만족 정도가 좋은 기법 일 수록 디스크 액세스 횟수가 적을 것이다. 본 연구에서는 각각의 색인 기법들이 어느 정도의 공간적 근접성을 보장하고 있는지를 분석하고 이를 기반으로 각 기법간의 성능을 비교하고자 한다. 기법간 성능 분석을 하기 위한 비교 과정은 다음과 같다.

단계 1] : 데이타 클러스터링 단계
집중분포로만 구성된 데이타들은 클러스터링 후 데이타를 각 클러스터 단위로 화일에 쓴다.

단계 2] : 트리 구성 단계
동일한 데이타를 각 공간 색인 기법을 사용하여 트리를 만든 후, 트리의 데이타 노드가 가리키는 데이타를 페이지 단위로 화일에 쓴다.

단계 3] : 맵핑 테이블 구성 단계
단계 1과 2에서 생성된 화일들을 각각 데이타 id를 기준으로 정렬한 후, 각 기법에 대해 데이타 id를 기준으로 페이지 번호와 클러스터 번호를 맵핑하여 맵핑 테이블을 만든다.

단계 4] : 비교 단계
테이블을 조사하여 한 페이지별로 맵핑되는 상이한 클러스터가 몇 개인지 조사하여, 평균 갯수를 구한다.

이와 같은 과정을 R-tree에 적용한 예를 [표 1]에서 [표 3]까지 나타내고 있다.

$$E(C) = \frac{1}{n} \sum_{i=0}^{\infty} C_i$$

여기서 $E(C)$ 는 전체 페이지에 대한 평균 클러스터 수

C_i 는 한 페이지내 상이한 클러스터 수

n 은 전체 페이지 수

여기서 $E(C_i)$ 결과 값으로 색인 기법간의 성능을 평가 할 수 있다. 즉 $E(C_i)$값이 클 수록 공간 근접성이 나쁜 경우 이며, 값이 1에 가까울 수록 공간 근접성이 좋은 경우이다. $E(C_i)$ 값이 1이라는 것은 한 페이지내 속한 데이타들이 모두 같은 클러스터내에 속하는 경우로써 가장 좋은 경우이다.

5 실험 결과

본 장에서는 4장의 단계 4에서 생성된 맵핑 테이블을 토대로 각 기법들의 공간적 근접성 정도를 계산하고, 계산 결과에 기반하여 분석하고자 한다. 실험 결과 분석에 앞서서 고려 할 사항으로는 데이타 클러스터링은 데이타가 고르게 분포되어 있는 상태에서 보다는 집중되어 있는 경우에 그 의미를 가지게 된다는 점이다. 본 실험에서는 집중분포만 구성된 데이타를 실험 데이타로써 사용하고 있으며, 집중 분포되어 있는 경우를 데이타 크기를 달리 하며 수행하고 있다. 한 페이지 크기는 한 페이지

[표 1] 단계 1 : 데이타 클러스터링

Cluster No.	ID	x	y
1	10	4260	1639
1	11	4044	1575
1	12	4068	1559
1	13	4278	1607
2	6	3838	6929
2	7	3902	6609
2	8	3982	6705
2	9	3790	6977
3	1	1222	1665
3	2	978	1089
3	3	1038	1361
3	4	1390	1081
3	5	1526	1625

[표 2] 단계 2 : R-tree구성 단계

Page No.	ID	x	y
3	10	4260	1639
3	11	4044	1575
4	12	4068	1559
2	13	4278	1607
2	6	3838	6929
2	7	3902	6609
2	8	3982	6705
2	9	3790	6977
1	1	1222	1665
1	2	798	1089
1	3	1038	1361
1	4	1390	1081
1	5	1526	1625

[표 3] 단계 3 : 맵핑 테이블 구성

Page No.	Cluster No.	ID
1	3	1
1	3	2
1	3	3
1	3	4
1	3	5
2	2	6
2	2	7
2	2	8
2	2	9
2	2	13
3	2	12
3	1	11
4	1	10

내 저장 가능한 객체 수가 비슷하도록 R-tree와 R'-tree는 1Kbytes로 R*-tree는 2Kbyte로 두었는데, R*-tree의 페이지 크기를 2Kbyte로 둔 것은 R*-tree구현시 데이타 노드에 저장하는 정

보가 데이타 외의 부가적인 정보들은 R-tree나 R'-tree 보다 더 많이 가지고 있기 때문이다. 하나의 클러스터 내에 저장 한 수 있는 데이타 갯수는 데이타 갯수가 1000, 2000개 인 경우는 200개 까지 저장 가능하며, 3000개 일 경우는 300개 까지 저장 가능하다. 아래의 [표 1]는 각각의 페이지에 대하여 한 페이지에 매핑되는 상이한 클러스터의 수를 모두 더한 후, 전체 페이지 수로 나눈 값으로, 전체 페이지에 대한 평균 클러스터 수를 구한 값이다. [표 1]은 집중적으로 분포되어 있는 경우이다.

[표 4] 데이타가 집중적으로 분포되어 있는 경우

색인기법 / 데이타 크기	R-tree	R+-tree	R*-tree
1,000개	1.8529	1.2667	1.1461
2,000개	2.5417	2.3448	1.2000
3,000개	3.0000	2.5164	1.2431

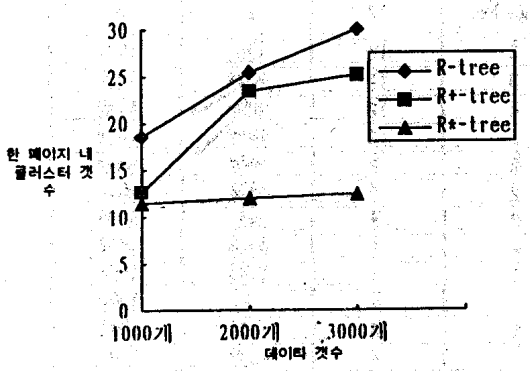

그림 2 데이타 크기에 따른 한 페이지 내 클러스터 수

[표 4]에서 나타난 결과는 R*-tree가 공간의 근접성을 다른 두가지 방법에 비해 훨씬 만족하고 있다는 것을 의미한다. 특히, R*-tree는 공간객체의 갯수에 대해 상당히 안정적이라는 사실을 나타낸다. 즉, 객체의 수에 거의 무관하게 근접성을 반영하고 있다는 것이다.

6. 결론 및 연구 방향

본 논문에서는 지리 데이타는 공간적 근접성을 기본으로 하여 검색된다는 사실에 따라 이를 공간 색인 기법들을 비교 분석하는 척도로 삼아, 다차원 공간 색인 기법인 R-tree와 R'-tree R*-tree들을 비교, 분석 하고 있다. 실험 결과에서도 나타난 것처럼 본 연구에서 비교 대상이 된 색인 기법들은 모두 매우 좋은 공간적 근접성을 만족하고 있으며, 데이타 검색시 나타나는 약간의 성능 차이는 공간적 근접성 문제보다는 구현한 시스템이 얼마나 안정적으로 수행되는가에 달려 있음을 알 수 있었다.

향후 연구 과제로써는 본 연구에서는 같은 R-tree계열을 비교 대상으로 삼고 있으나, 일차원 인덱싱 기법인 Grid-file 이나 Quadtree 등 다양한 색인 기법들을 대상으로 해서 같은 비교 방법을 적용하는 것이 가능 할 것이라 본다. 두번째로는 본 연구에서 제안한 데이타 클러스터링과 각 공간 색인 기법에 결합하으로써 각 색인 기법들이 최적의 공간 근접성을 만족하으로써 보다 빠른 속도로 데이타를 검색 가능하도록 하는 것이다.

마지막으로 질의 처리시 필요한 데이타들을 디스크에서 가지오는 작업이 수행되어야 하는데 이러한 디스크 액세스 작업을 순차적으로 수행하는 것이 아니라, 자주 함께 사용되는 데이타들을 클러스터링 한 후, 몇 개의 디스크에 분산 저장하여 Parallel R-tree로 디스크 액세스 자체를 병렬화 함으로써 데이타 검색 속도를 감소 시킬 수 있을 것이다.

7 참고 문헌

[1] O. Gunether and A.Buchmann, "Research Issues is Spatial Databases", SIGMOD Record. Vol.19 No.4, 1990, pp. 61-68

[2] Hongjun Lu and Beng-Chin Ooi, "Spatial Indexing : Past and Future", IEEE Data Engineering, Vol.16 No.3, 1993, pp. 16-22

[3] E. G. Hoel and H. Samet, "A Qualitative Comparison Study of Data Structure for Large Line Segment Databases", Proc.SIGMOD'92, 1992, pp.205-214

[4] Ki-Joune Li and Robert Laurini, "The Spatial Locality and a Spatial Indexing Method by Dynamic Clustering Method is Hypermap System", 2nd Conf.SSD, Advanced Spatial Notes on Conputer Science No 525, 1991, pp.207-223

[5] T. Brinkhoff, H-P. Kriegel and B. Seeger, "Efficient Processing of Spatial Joins Using R-trees", Proc.SIGMOD'93, 1993, pp.237-246

[6] T. Sellis, N. Roussopoulos and C. Faloutsos, "The R+-tree : A Dynamic Index for Multidimensional objects", Proc.VLDB 1987, pp.507-518

[7] N. B, H-P. Kriegl and B. Seeger, "The R*-tree : An Efficient and Robust Access method for Points and Rectangles", Proc.SIGMOD'90, pp.322-331

[8] E. Omiecinski, L. Lee and P. Scheuermann, "Perforamnce Analysis fo a Concurrent File Reorganization Algorithm for Record Clustering", IEEE Knowledge and Data engineering, Vol. 6 No. 2, APRIL 1994, pp.248-257