
한정된 저장 공간상에서 이동 객체 궤적들에 대한 개선된 분할 알고리즘에 관한 연구

박주현* · 조우현**

A Study on Improved Split Algorithms for Moving Object Trajectories in Limited Storage Space

Ju-Hyun Park* · Woo-Hyun Cho**

이 논문은 2008년도 부경대학교의 지원을 받아 수행된 연구임(PK-2008-030)

요 약

무선 네트워크 기술의 발달로, 지속적으로 위치가 변화하는 시공간 오브젝트의 위치 정보는 다양한 어플리케이션에서 사용되고 있다. 이런 시공간 오브젝트는 많은 위치 정보를 가지고 있지만 이 오브젝트의 모든 궤적 정보를 저장한다는 것은 비효율적이다. 이것은 저장 매체의 저장 공간은 한정되어 있기 때문이다. 따라서 본 논문에서는 제한된 저장 공간에서 효율적으로 궤적을 분할하는 방법을 제시한다. 개선된 분할 알고리즘을 바탕으로, MBR들의 면적을 최소화 하는 궤적들을 분할하는 k split 알고리즘을 제시한다. 실험의 결과로 제시하는 분할 방법이 다른 알고리즘보다 더 효율적인 것을 알 수 있다.

ABSTRACT

With the development of wireless network technology, the location information of a spatiotemporal object which changes their location is used in various application. Each spatiotemporal object has many location information, hence it is inefficient to search all trajectory information of spatiotemporal objects for a range query. In this paper, we propose an efficient method which divides a trajectory and stores its division data on restricted storage space. Using suboptimal split algorithm, an extended split algorithm that minimizes the volume of EMBRs(Extended Minimum Bounding Box) is designed and simulated. Our experimental evaluation confirms the effectiveness and efficiency of our proposed splitting policy

키워드

궤적 분할, 이동 객체, 시공간 색인, 범위 질의

Key word

Trajectory Splitting, Moving Object, Spatiotemporal Indexing, Rang Query

* 부경대학교 컴퓨터공학과
** 부경대학교 컴퓨터공학과 (교신저자, whcho@pknu.ac.kr)

접수일자 : 2010. 07. 27
심사완료일자 : 2010. 08. 05

I. 서 론

최근 이동통신, 무선 네트워크, GPS(Global Position System)의 기술적인 발전으로 인하여 다양한 공간상에서 움직이는 객체들의 위치 정보를 수집하여 실시간으로 정보를 제공하는 위치 기반 서비스가 점점 늘어나고 있는 추세이다. 따라서 이러한 움직이는 객체들의 위치 정보를 수집하고 저장하고 관리하기 위한 이동 객체 데이터베이스(Moving Object Database)에 대한 관심이 증가하고 있고 이러한 이동 객체를 효율적으로 찾기 위한 이동 객체를 위한 인덱스 기법에 관한 연구도 활발하게 진행되고 있다. 현재까지 대부분의 색인 방법들은 최소 경계 사각형(Minimum Bounding Rectangle)과 같이 실제 데이터에 대한 근사치 영역을 사용하여 색인을 구성한다.

이동 객체의 색인을 위한 기존의 방법들을 살펴보면 첫 번째로 질의 종류에 따라 이동 객체의 이력정보를 검색하기 위한 방법[2, 4-6] 두 번째로 이동 객체의 궤적을 검색하기 위한 방법[1, 3, 8, 9, 13] 세 번째로 이동 객체의 궤적을 분석하여 미래를 예측하기 위한 색인 방법[10, 11] 이렇게 크게 3가지로 나눌 수 있다. 본 논문에서는 두 번째 방법인 이동 객체의 궤적을 검색하기 위한 방법에 초점을 맞춘다. 이동 객체의 궤적이란 공간상에서 시간의 변화에 따라 움직이는 객체의 이동경로, 시간의 흐름에 따른 포인트의 순서라고 할 수 있다. 규칙적인 시간 혹은 불규칙적인 시간동안 움직인 포인트와 포인트 사이를 하나의 영역으로 확장을 하는데 이것이 최소 경계 사각형이 된다. 이러한 궤적 기반의 데이터를 검색하기 위한 질의 방법에는 두가지가 있는데 좌표 기반 질의와 궤적 기반 질의가 있다.[1] 첫 번째 좌표 기반 질의는 주어진 시간 간격 동안 어떠한 공간을 지나가는 궤적의 객체 식별자와 궤적의 객체의 포인트들을 반환한다. 이렇게 반환된 궤적의 객체의 포인트들을 최소 경계 사각형으로 변환하여 입력된 공간 질의에 포함이 되는지를 찾게 된다. 두 번째 궤적 기반 질의는 궤적들이 서로 복잡하게 교차하는 위상적인 관계 혹은 주어진 시간 간격 동안 어느 공간 안에서의 속도변화와 이동방향등의 정보를 반환하게 된다.

이 역시 위상적인 관계나 속도변화와 이동방향등의 정보를 이용하여 최소 경계 사각형으로 구성하여 공

간 질의에 포함되는지를 찾게 된다. 이처럼 궤적 기반의 데이터를 검색하기 위한 방법에서는 공간 질의를 처리하기 위해서 궤적을 분할하여 MBR들을 생성하고 이를 이용하여 색인을 생성하는 것이 공간 질의를 처리하기 위해서는 아주 중요하다. 하지만 어떤 시공간상의 궤적들과 그 궤적이 갖는 객체의 포인트들은 무수히 많다. 이러한 궤적의 모든 정보를 찾는 것은 많은 시간이 소요된다. 따라서 본 논문에서는 검색 시간을 단축하기 위한 효율적인 색인을 생성하기 위해 저장 공간상의 궤적의 분할의 개수를 제한하는 알고리즘을 제안한다.

제안하는 방법은 이전에 제안한 분할과정에서 질의 범위의 크기를 고려한 확장된 최소경계사각형(EMBR : Extended MRR)의 면적을 최적으로 가깝게 최소화는 알고리즘인 개선된 분할(Improved Split) 알고리즘[14]을 적용하여 분할의 개수가 k 개로 제한할 경우 EMBR의 총면적이 가장 최소에 가까운 분할을 만들어 내어 색인 구성 후 질의 수행 과정에서 불필요한 탐색 공간을 감소시킬 수 있는 효율적인 분할 알고리즘인 k split 알고리즘을 제안한다. 성능 평가 결과로 병합분할방법(MergeSplit)[12]보다 EMBR의 총합에 있어서 더 최적의 분할이 된다는 것을 알 수 있었다. 분할의 위치를 찾는 데 우선 순위 큐(Priority-Queue)[12]를 사용하여 계산에 소요되는 시간을 단축시켰다. 본 논문에서 제안하는 방법은 궤적의 수가 많을 경우에 분할의 수가 제한적일 경우에도 확장이 용이하다.

본 논문의 구성은 다음과 같다. 2절에서는 이동객체의 시공간 색인에 대한 관련 연구와 본 논문에서 사용된 개선된 분할 알고리즘에 관한 설명하고 3절에서는 개선된 분할 알고리즘에 의해 생성된 MBR의 집합을 분할의 개수를 k 로 제한하여 색인 검색 시간을 단축하기 위해 비교검색영역을 최소가 되게 하는 k 알고리즘에 관해 제안한다. 4절에서는 제안하는 알고리즘의 우수성을 입증하기 위해 기존의 알고리즘과의 성능 평가를 비교 분석한다. 5절에서는 결론 및 향후 연구 과제에 대해 기술한다.

II. 관련 연구

1. EMBR

본 연구에서는 이동 객체 궤적의 위치 정보를 간단하게 나타내기 위해서 위치 정보를 1차원 좌표로 표현하고 시간 정보를 고려해서 궤적을 2차원 공간으로 나타낸다.

EMBR면적의 계산은 그림 1과 같이 범위질의 가로 $qx/2$, 세로 $qt/2$ 만큼 MBR에서 확장된 사각형 면적을 계산하면 된다. EMBR은 MBR의 질의 수행 과정에서 범위 질의가 주어졌을 때 질의가 근사치 영역 MBR과 중첩하는지 여부를 찾아내기 위하여 비교 검색해야 하는 부분을 나타낸 것이다. 그러므로 질의의 중심점에서 각 면까지의 길이만큼 MBR의 각 면에서부터 확장시킨 영역을 나타낸다.

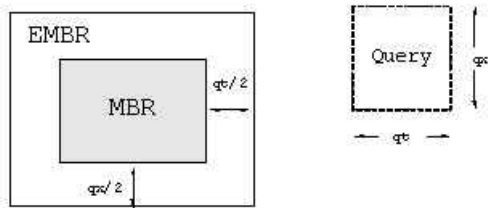


그림 1. 범위질의 크기를 고려하여 확장된 MBR
Fig 1. Size of range query considered extended MBR

2. 개선된 궤적 분할 알고리즘

개선된 궤적 분할 알고리즘(Improved split algorithm)은 그림 4와 같다. 그림 5의 알고리즘을 간략히 설명하면 우선 P_0 에서 P_n 까지 위치 포인트들로 이루어진 궤적과 평균적인 질의의 크기가 주어진다. 첫 번째 과정 1)에서 그림 2와 같이 전체 분할 MBR집합을 만들고 두 번째 과정 2)에서 키 값을 계산해서 차례대로 우선순위 큐에 삽입한다. 다음 세 번째 과정 3)에서 정렬된 우선순위 큐에서 첫 번째로 반환되는 키 값 $KEY_{(i,j)}$ 을 이용해 MBR집합을 갱신하고 우선순위 큐를 재구성한다. 음의 수를 가지는 키 값이 더 이상 존재하지 않을 때까지 3)의 과정을 반복 수행한 후 그림 3과 같이 최종적인 MBR집합을 만들어 낸다.

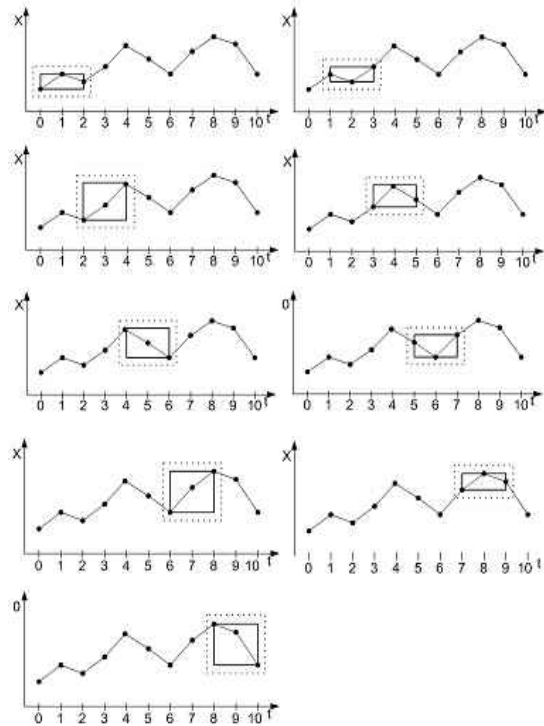


그림 2. 인접한 두 개의 MBR을 병합하여 만든 새로운 MBR
Fig 2. New MBR combined close two MBR

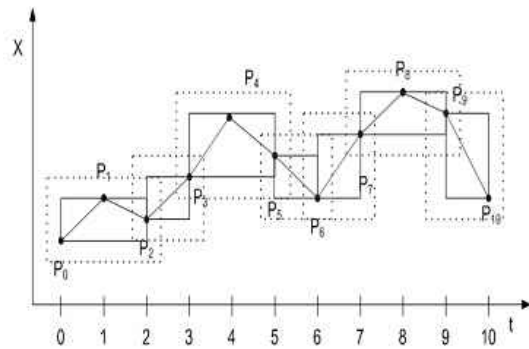


그림 3. 개선된 궤적 분할 방법으로 구성된 최종 MBR의 집합
Fig 3. Set of MBR constituted the improved split algorithm

[정리] Algorithm-1의 시간 복잡도는 $O(n \log n)$ 이다.
증명.

1)에서 전체 분할 MBR집합을 만들기 위한 연산 횟수는 n 이고, 2)에서 하나의 키 값을 계산하여 우선순위 큐에 삽입하는 연산 횟수는 $1 + \log n$ 이며 총 n 번 수행이 되므로 $n + n \log n$ 이다. 3)에서 MBR집합의 갱신과 우선순위 큐의 재구성을 위한 연산 횟수는 $n \log n$ 이다. 그래서 알고리즘의 총 연산 횟수는 $n + n + n \log n + n \log n$ 이다. 그러므로 이 알고리즘의 시간 복잡도는 $O(n \log n)$ 이다.

```

Algorithm-1 : Improved Split for a Trajectory
Input: A set of Trajectory's Points( $P_0, P_1, \dots, P_n$ ),
Query size.
Output: A set of MBRs that cover the Trajectory.
1) Construct full split MBRs  $M_i[0, n]$ 
2) For each  $i$  ( $0 \leq i \leq n-1$ )
   Compute EMERs volume for merging  $M_{[i, i+1]}$ 
   with  $M_{[i+1, i+2]}$ 
   store  $KEY_{(i, i+2)}$  in a priority queue such
   that follow holds:
    $KEY_{(i, i+2)} = E_{[i, i+2]} - (E_{[i, i+1]} + E_{[i+1, i+2]})$ 
3) While ( $KEY_{(i, j)}$  in the root node of the priority
   queue is negative)
   Use the priority queue to merge the pair of
   consecutive  $M_i[0, n]$  that give the biggest
   decrease in EMER volume.
   Update the priority queue with new merged MBRs.
    
```

그림 4. 개선된 궤적 분할 알고리즘
Fig 4. Improved split algorithm

III. 분할의 개수를 k로 제한하는 궤적분할 알고리즘

본 절에서는 제한된 저장 공간상에서 이동 객체의 궤적에 대한 색인 과정에서 적절한 분할 위치를 찾아 MBR들을 만들기 위한 k 분할 알고리즘을 제안한다.

1. 분할의 개수를 k개로 제한하는 알고리즘의 분할 과정

공간을 나타내는 평면상에 $P_0 \sim P_n$ 까지 이동 객체 위치 포인트들로 이루어진 궤적이 기록되어 있다. 첫 번째 과정에서 이 궤적에 개선된 분할 알고리즘을 적용하면 최적에 가까운 분할을 만들어 낸다. 두 번째 과정으로 위치 포인트의 개수를 k 개로 줄이는 분할을 하기 위해서 우선순위 큐에 제일 상위에 랭크된 키 값을 반환하여 키 값에 들어있는 두 개의 위치 정보 i 와 j 를 가져온다. 단 우선순위 큐에 있는 음의 수를 가지는 키 값은 개선된 분할 알고리즘에 의해 더 이상 존재하지 않기 때문에 양의 수를 가지는 키 값들 중에 제일 작은 값을 사용한다. 그리고 i 와 j 로 만들어지는 $M[i, j]$ 를 사용하여 분할 개수가 하나 줄어든 새로운 $MBR_{l-1}[0, l]$ 으로 갱신한다. 그리고 반복문의 다음 루프에서 갱신에 사용될 합병된 MBR을 찾기 위해 우선순위 큐를 재구성한다. 재구성 과정은 개선된 분할 방법과 같다. 두 번째 과정은 위치 포인트의 개수 i 이 k 개가 될 때까지 반복 수행한다. 한번의 수행 과정이 진행될 때마다 MBR의 총면적이 최대로 감소하는 합병된 MBR이 선택되어 MBR집합의 갱신에 사용된다.

2. 분할의 개수를 k로 제한하는 알고리즘의 궤적 분할 알고리즘

분할의 개수를 k 개로 제한하는 궤적 분할 알고리즘은 그림 5의 알고리즘을 이용하여 그림 5와 같이 간단히 설계할 수 있다. 그림 4의 알고리즘을 통해 구성된 l 개의 분할이 k 개보다 더 작다면 그림 5의 알고리즘이 수행될 필요가 없다. 하지만 그림 4를 통해 만들어진 분할의 개수 i 가 k 보다 많다면 우선순위 큐에 정렬된 첫 번째 요소의 키 값을 이용하여 MBR집합의 갱신과 우선순위 큐의 재구성 작업을 i 가 k 가 될 때까지 계속 수행한다. 이 과정에선 MBR의 합병으로 인한 EMER 면적의 변화가 최소로 증가되는 MBR 묶음을 갱신작업에 사용한다.

Algorithm-2 : Split with k splits for a Trajectory
Input: A set of Trajectory's Points(P_0, P_1, \dots, P_n), Query size.
Output: A set of MBRs with k splits that cover the trajectory.

1) Construct a set of MBRs using Algorithm-1
 3) While ($i > k$)
 Use the priority queue to merge the pair of consecutive $M_i[o, n]$ that give the smallest increase in EMBR volume;
 Update the priority queue with new merged MBRs;
 $i = i + 1$

그림 5. 분할 개수를 k개로 제한하는 궤적 분할 알고리즘
 Fig 5. k split algorithm

[정리 2] Algorithm-2의 시간복잡도는 $O(n \log n)$ 이다. 증명.

첫 번째 단계 1)에서 Algorithm-1의 연산 횟수는 $2n + 2n \log n$ 이다. 두 번째 단계 2)에서 분할 개수 i 이 k 가 될 때까지의 MBR 집합의 갱신과 우선순위 큐의 재구성을 위한 연산 횟수는 $n \log n$ 이다. 그래서 총 연산 횟수는 $2n + 3n \log n$ 이다. 그러므로 이 알고리즘의 시간복잡도는 $O(n \log n)$ 이다.

IV. 실험 및 성능 비교

본 논문에서 제안하는 한정된 저장공간상에서 이동 객체 궤적들에 대한 개선된 분할 방법의 성능실험은 펜티엄-IV 3.2GHz 프로세서와 메모리 1.5Gbyte, Windows XP 운영체제를 사용하는 시스템상에서 수행했으며, 알고리즘의 구현을 위해서 C++ 언어를 사용하였다. 그리고 성능 비교 평가를 위하여 기존의 합병 분할 방법과 k split 알고리즘을 구현하여 실험을 하였다. 본 실험에서는 측정 시간과 위치 정보로 표현된 2차원의 좌표 상에서 위치 축의 총범위는 0에서 1500으로 가정하였고 포인

트의 개수는 1000에서 2000으로 가정하였다. 그리고 실험 데이터의 객관성을 위하여 한계 조건내의 각 데이터를 임의 생성하여 실험에 적용하였다.

첫 번째 실험에서 규칙적인 시간의 간격에 따라 변화는 이동 객체의 위치 정보를 1000개에서 2000개까지 200개 단위로 생성하였다. 생성 과정에서 위치 변화의 정도는 각각 15, 30, 45로 한계를 설정하고, 평균 범위 길이의 크기를 30으로 고정하고 위치 포인트의 저장 공간을 700개로 한정하여 각 분할 방법을 통해 줄어드는 MBR 집합의 EMBR 총면적(Sum of EMBR)을 계산하여 비교하였다.

그림 6-8은 이동 객체의 위치 변화 정도를 다양하게 하며 실험한 각 궤적 분할의 결과이다. 성능 비교 결과를 보면 본 논문에서 제안하는 한정된 저장 공간상의 개선된 분할 방법을 통해 만들어지는 MBR 집합의 EMBR 총면적이 합병 분할 방법을 통해 만들어지는 MBR 집합의 EMBR 총면적보다 더 효율적으로 줄어든다는 것을 알 수 있다.

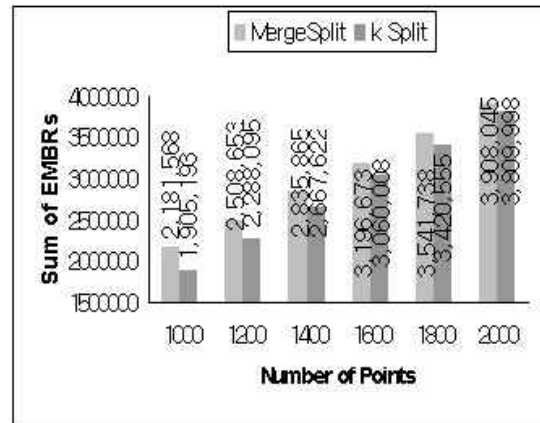
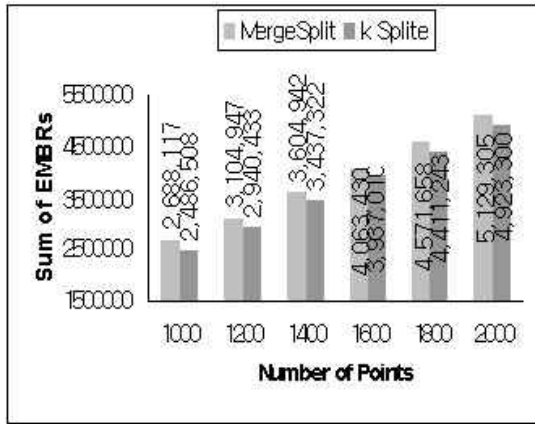
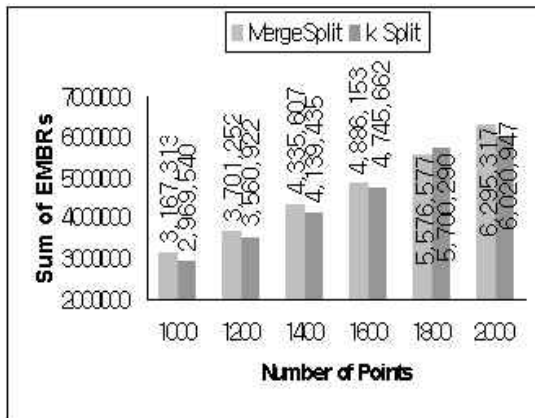


그림 6. 위치정보에 대한 측정시간이 규칙적인 경우 (위치변화 정도 15이내, 측정시간 구간 35)

Fig 6. The regular case of measurement time of location information



(위치변화 정도 30이내, 측정시간 구간 35)
 그림 7. 위치정보에 대한 측정시간이 규칙적인 경우
 (location change within 30, measurement time interval 35)
 Fig 7. The regular case of measurement time of location information

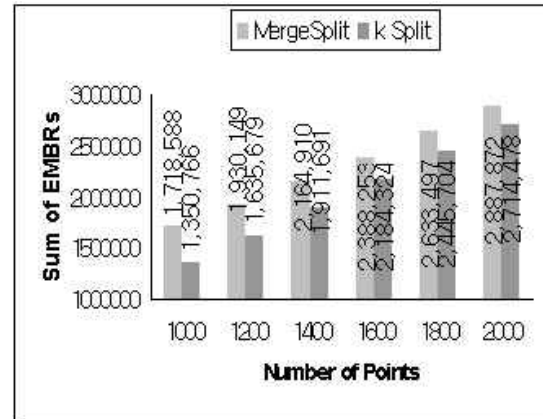


(위치변화 정도 45이내, 측정시간 구간 35)
 그림 8. 위치정보에 대한 측정시간이 규칙적인 경우
 (location change within 45, measurement time interval 35)
 Fig 8. The regular case of measurement time of location information

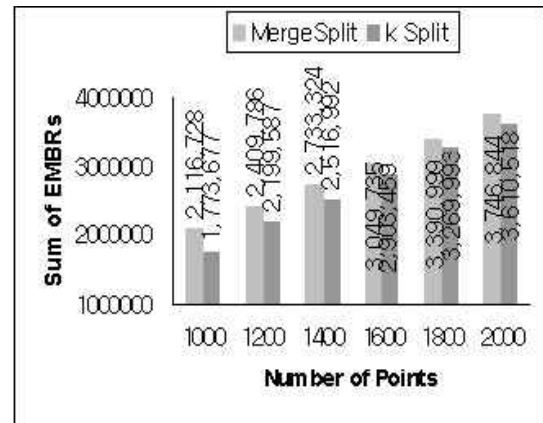
두 번째 실험에서는 시간 구간을 10에서 35 사이에서 변화를 주어가며 불규칙적으로 이동 객체의 위치 정보를 생성하여 실험에 적용하였다. 그 외의 조건은 첫 번째 실험과 같은 조건을 주었다.

그림 9-11는 두 번째 실험을 통한 각 궤적 분할 방법의 성능 비교 결과이다. 첫 번째 실험과 마찬가지로 제

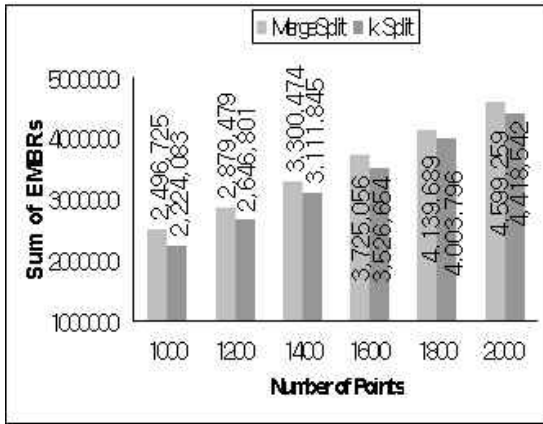
한하는 궤적 분할 방법을 통해 만들어지는 MBR집합의 EMBR의 총면적이 합병 분할 방법의 MBR집합의 EMBR의 총면적보다 더 효율적으로 줄어들음을 알 수 있다.



(위치변화 정도 15이내, 측정시간 구간 10-35)
 그림 9. 위치정보에 대한 측정시간이 불규칙적인 경우
 (location change within 15, measurement time interval 10-35)
 Fig 9. The irregular case of measurement time of location information



(위치변화 정도 30이내, 측정시간 구간 10-35)
 그림 10. 위치정보에 대한 측정시간이 불규칙적인 경우
 (location change within 30, measurement time interval 10-35)
 Fig 10. The irregular case of measurement time of location information



(위치변화 정도 45이내, 측정시간 구간 10-35)
 그림 11. 위치정보에 대한 측정시간이 불규칙적인 경우
 (location change within 45, measurement time interval 10-35)

Fig 11. The irregular case of measurement time of location information

V. 결 론

본 논문에서는 이동 객체의 궤적 정보가 저장 공간이 제한되어 있는 상태에서 궤적 정보를 효율적으로 줄이는 알고리즘을 제안하였다. 제안하는 궤적 분할 알고리즘은 평균적인 질의의 크기를 이용한 EMBR 면적을 우선 고려하여 궤적의 적절한 분할 위치를 찾는다. 성능 비교 결과 제안하는 궤적 분할 방법이 기존의 병합 분할 방법보다 EMBR 면적을 줄여줌으로서 더 효율적으로 궤적을 분할한다는 것을 알 수 있었다. 결과를 보면 제안하는 궤적 분할 방법을 이용하여 이동 객체의 궤적에 대한 색인을 구성해 둔다면 저장 공간이 한정되어 있는 상황에서 더 효율적으로 색인을 사용할 수 있고 질의 수행 과정에서 불필요한 탐색 공간을 줄임으로서 처리하는 데이터의 양을 줄일 수 있음을 알 수 있다.

향후 논문 과제는 더욱 다양한 궤적들에 대하여 색인 구성 과정에서 제한된 저장 공간상에서 분할 개수를 한계를 두어 이동 객체 궤적을 정보를 효율적으로 저장할 수 있도록 하는 알고리즘에 대한 실험을 수행할 예정이다. 더 나아가 포인트와 포인트 사이의 궤적의 길이가 긴 궤적을 효율적으로 분할하는 알고리즘에 관한 연구를 수행할 예정이다.

참고문헌

[1] D. Pfoser, C. S. Jensen and Y. Theodoridis, "Novel Approaches to the Indexing of Moving Object Trajectories", Proc of The 26th International Conference on Very Large Data Bases, pp. 395-406, 2000

[2] X. Xu, J. Han and W. Lu, "RT-Tree: An Improved R-Tree Indexing Structure for Temporal Spatial Databases", Proc of The International Symposium on Spatial Data Handling, pp. 1040-1049, 1990

[3] Y. Theodoridis, M. Vazirgiannis and T. Sellis, "Spatio-Temporal Indexing for Large Multimedia Application", Proc of The 3rd IEEE International Conference on Multimedia Computing and Systems, pp. 441-448, 1996

[4] O. Wolfson, S. Chamberlain, S. Dao and L. Jiang, "Location Management in Moving Objects Databases", Proc of The International Workshop in Satellite-Based Information Services, pp. 7-14, 1997

[5] T. Brinkhoff, "Generating Network-Based Moving Objects", Proc of The International Conference on Scientific and Statistical Database Management, pp. 253-255, 2000

[6] R. Benetis, C. S. Jensen, G. Karciuskas and S. Saltenis, "Nearest Neighbor and Reverse Nearest Neighbor Queries of Moving Objects", Proc of The International Symposium on Database Technology, pp. 44-53, 2002

[7] D. E. KNUTH, "The Art of Computer Programming, Volum3, Sorting and Searching Second Edition", Addison Wesley Longman

[8] Z. Song and N. Roussopoulos, "SEB-tree: An Approach to Index Continuously Moving Objects", Proc of The 4th International Conference on Mobile Data Management, pp. 340-344, 2003

[9] V. P. Chakka, A. C. Everspaugh and J. M. Patel, "Indexing Large Trajectory Data Sets With SETI", Proc of The 1st Biennial Conference on Innovative Data Systems Research, 2003

- [10] S. Prabhakar, Y. Xia, D. V. Kalashnikov, W. G. Aref and S. E. Hanbrusch, "Query Indexing and Velocity Constrained Indexing: Scalable Techniques for Continuous on Computers", IEEE Transactions on Computers, pp. 1124-1140, 2002
- [11] Y. Tao, D Papadias and J. Sun, "The TPR*-Tree: An Optimized Spatio-Temporal Access Method for Predicative Queries", Proc of the 29th International Conference on Very Large Databases, pp. 790-801, 2003
- [12] M. Hadjieleftheriou, G. Kollios, V. J. Tsotras, D. Gunopulos, Efficient Indexing of Spatiotemporal Objects, Lecture Notes in Computer Science, pp. 251-268
- [13] H. Zhu, J. Su, O. Ibarra, "Trajectory Queries and Octagouse in Moving Object Databases", Proc of The ACM International Conference on Information and Knowledge Management, pp. 413-421, 2002
- [14] Hyun-Jun Jeon, Ju-Hyun Park, Hee-Suk Park, Woo-Hyun Cho, "An Improved Split Algorithm for Indexing of Moving Object Trajectories" KIPSTD, 16-D, pp. 161-168, 2009



조우현(Woo-Hyun Cho)

1985년 경북대학교 전자공학과
전산 공학전공(공학사)
1988년 경북대학교 대학원
전자공학과 (공학석사)

1998년 경북대학교 대학원 전자공학과(공학박사)
1989년-현재 부경대학교 컴퓨터공학과 교수
※관심분야: 지능형 데이터베이스, 멀티미디어
인텍싱, 객체 데이터베이스 관리 기술

저자소개



박주현(Ju-Hyun Park)

2000년 경남대학교 컴퓨터공학과
(공학사)
2002년 경남대학교 대학원
컴퓨터공학과(공학석사)

2007년 부경대학교 컴퓨터공학과 (박사수료)
※관심분야: 공간 데이터베이스, 객체지향 데이터
베이스