

# 전자태그 객체의 추적을 위한 간격 R-트리의 분할 정책

이세호<sup>o</sup> 안성우 홍봉희 반재훈\* 임덕성\*\*

부산대학교 컴퓨터 공학과<sup>o</sup>, 경남정보대학 인터넷응용계열\* 영진전문대학 컴퓨터정보계열\*\*  
{rustyangel<sup>o</sup>, swan, bhong}@pusan.ac.kr, chban@kit.ac.kr\*, junsung@yjc.ac.kr\*\*

## Splitting polies of interval R-Trees for tracking RFID tag objects

Seho Lee<sup>o</sup>, Sungwoo Ahn, Bonghee Hong Chaehoon Ban\*, Duksung Lim\*\*

Dept. of Computer Engineering, Pusan National University<sup>o</sup>

Kyungnam College of Information & Technology\*

Division of Computer information Technology, Yeungin College\*\*

### 요 약

RFID(Radio Frequency Identification)는 자동 인식 데이터 수집 장치의 한 분야로써 GPS장치를 장착한 차량과 같이, RFID 태그(전자태그)를 상품에 부착하여 인식, 추적할 수 있다. 이러한 전자태그 객체는 시간에 따라 경로 정보가 누적되는 이동체와 유사한 특성을 가진다. 그러나 기존의 이동체 색인과 달리 태그 객체의 위치는 판독기의 위치로 인식되어 위치보고가 판독기의 인식영역 안에서만 이루어지므로 보고 주기를 예측할 수 없다. 기존의 이동체 색인에서 전자태그의 특성을 표현하기 힘들기 때문에 전자태그의 특성을 고려한 색인이 필요하게 되었다. 최근 전자태그의 특성을 고려한 색인인 TPIR-Tree(Time parameterized Interval R-Tree)가 발표되었다. 그러나 이 색인은 기존 공간 색인에서의 균등분할 기법을 사용하여 위치보고가 시간의 순서를 가지는 전자태그의 특성을 고려하지 못하여 과거노드의 저장효율이 좋지 못하다.

이 논문에서는 TPIR-Tree의 저장효율 및 검색 성능 향상을 위해서 시간의 순서에 따라 위치를 보고하는 전자태그 객체의 특성을 고려하여 분할축 선정 기법 및 시간축 분할시 비균등 분할정책을 제안한다.

### 1. 서 론

RFID(Radio Frequency Identification)[1]는 무선 주파수를 이용하여 전자태그를 부착한 객체를 판독기가 자동으로 인식하고 확인하는 기술로써 최근 유통환경에서 RFID를 사용하는 응용분야가 주목받고 있다. 이러한 기술은 재고 관리, 물류의 공급망 관리, 공장 자동화등과 같은 다양한 응용분야에서 사용되고 있다. 전자태그 객체는 기존 이동체와 비교하여 다음과 같은 차이점을 보인다. 첫째, 전자태그 객체의 위치보고는 비주기적이다. 전자태그의 경우 판독기의 인식 범위에 들어 갈 때 보고하게 되므로 위치보고 주기를 알 수 없다. 둘째, 기존 이동체의 위치는 공간상 균등하게 분포함을 가정하지만, 전자태그의 경우 판독기의 위치를 전자태그의 위치로 인식하므로 판독기의 위치에 의존적이다. 셋째, 이동체는 시간이 지남에 따라 위치가 연속적으로 변하지만, 전자태그 객체는 판독기에 들어 갈 때와 나갈 때 위치를 보고하므로 이 시간 간격 동안 동일한 위치를 가진다. 따라서 전자태그 객체의 궤적은 시간축에 평행한 선분으로 표현된다. 넷째, 전자태그 객체의 궤적은 연결되어 있지 않다. 위치의 보고가 태그의 위치가 아닌 판독기의 위치를 보고하고, 위치보고 주기가 비주기적이므로 각 보고위치를 선분으로 연결할 수 없다. 다만 전자태그가 어떤 판독기에 들어 갈 때와 나갈 때의 두 시점만을 연결할 수 있다.

RFID환경에서의 위치추적기술이 많은 관심을 받고 있지만 이러한 환경을 고려한 색인의 연구가 미진한 상태이다. 최근 RFID환경을 고려한 색인인 TPIR-Tree(Time parameterized Interval R-Tree)가 발표되었다[2]. 이 색인의 경우 판독기 안에 들어가서 위치정보를 보고하고 아직 판독기를 벗어나지 않아서 완성되지 않은 궤적은 Now Interval로 표현한다. 또한 데이터의 삽입과 노드의 분할 시 이를 부모노드의 시간 경계까지

시간 값을 확장해 처리하여 효과적인 데이터 처리와 현재 질의를 가능하게 하였다. 이 색인의 경우 삽입, 분할시 R\*-Tree의 방식을 사용하고 있다. 그러나 RFID환경에서 전자태그는 이동체와 같이 위치보고가 시간의 순서를 가진다. 즉, 태그의 삽입은 시간의 순서를 가지며, 과거의 노드에 새로운 데이터 삽입은 일어나지 않는다. 또한 태그의 궤적은 시간축에 대해서만 평행하게 생성되고 다른축에 대해서는 성장을 하지 않는다. 따라서 시간축 분할시 균등 분할을 할 경우 과거의 노드에는 더 이상의 삽입이 일어나지 않으므로 저장 공간의 활용도가 떨어진다. 본 논문에서는 태그 객체의 위치보고가 시간의 순서를 가지므로 시간축 분할 시 비균등 분할 기법을 통해 저장효율과 검색성능이 보다 효율적인 분할정책을 제안한다.

이 논문의 구성은 다음과 같다. 2장에서는 관련연구를 소개하고, 3장에서는 대상 환경 및 TPIR-Tree에서의 분할 시 문제점을 정의한다. 4장에서는 TPIR-Tree 노드에 대한 분할의 각 사례들을 분류한다. 5장에서는 보다 효율적인 분할정책을 제시한다. 마지막으로 6장에서는 결론 및 향후 연구를 기술한다.

### 2. 관련연구

이 장에서는 기존 시공간 색인과 TPIR-Tree[2] 대하여 기술한다. 기존의 시공간 색인에는 R\*-Tree[3], 시간 기반 R-Tree[4]등이 있다.

R\*-Tree의 경우 R-tree가 삽입 시에 영역만을 고려하는 단점을 보완하기 위하여 검침과 가장자리를 추가적으로 고려한 색인구조이다. 그러나 이동체의 특성상 현재 시간을 기준으로 객체의 삽입이 진행되기 때문에 시간축 분할시 균등 분할하게 되면 과거 노드로의 삽입이 더 이상 없으므로 공간 활용율이 떨어지게 된다.

시간 기반 R-tree의 경우 이동체의 특성을 고려하여 시간축 분할시 비균등 분할 기법을 사용한다. 하지만, 이 색인의 경우 이동체가 주기적으로 위치정보를 보고하고 이동체의 분포는 공

본 논문은 교육인적자원부 지방연구중심대학육성사업 "차세대물류IT기술연구사업단"의 지원에 의하여 연구되었음

간상에 균등 분포한다는 가정이 있으므로, 앞서 말한 전자태그의 위치보고가 비주기적이고 태그의 분포는 공간상에서 비균등 분포를 가지기 때문에 RFID 환경에 그대로 적용할 수가 없다.

TPIR-Tree의 경우 전자태그 객체의 특성인 단절된 궤적의 표현과 전자태그가 판독기 안에 머물러 있는 데이터를 Now Interval로 표현하여 처리하는 기법을 통해 데이터의 삽입시 노드 선택 기법과 현재 질의 처리 기법을 제시하였다. 그러나 전자태그 객체의 위치보고는 시간의 순서를 가지므로 과거의 시간에 대하여 위치보고가 없다는 특성을 고려하지 않았기 때문에 시간축 분할에 대해서도 균등분할을 수행하여 저장효율이 떨어진다.

3. 대상 환경 및 문제 정의

이 장에서는 RFID 시스템의 환경을 기술하고, 이를 고려한 TPIR-Tree에서의 분할 정책 및 문제점을 제시한다.

3.1 대상 환경

RFID 시스템은 판독기(reader)와 전자태그(RFID tag), 그리고 호스트서버로 구성된다. 판독기는 자신의 태그인식영역에 들어오거나 나가는 태그의 정보를 수집한다. 이때 태그가 인식영역에 들어오면 판독기는 태그에 대하여 Enter 이벤트를 발생하여 서버에 전송하고, 인식영역을 나갈 때는 Leave 이벤트를 발생하여 그 정보를 서버에 전송하게 된다.[1]

이동체에 대한 기존 연구에서는 이동체에 대한 궤적을 보고한 두 시점의 시공간 위치를 연결한 선분으로 표현한다. 즉, 이동체가  $t_i$ 시간에  $(x_i, y_i)$ 의 위치를 보고하고  $t_{i+1}$ 시간에  $(x_{i+1}, y_{i+1})$  위치를 보고하는 경우 이동체의 궤적은 보고한 두 개의 시공간 위치를 연결하는 선분인  $(x_i, y_i, t_i), (x_{i+1}, y_{i+1}, t_{i+1})$ 로 표현된다.[3]

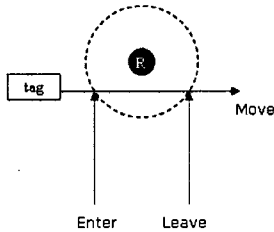


그림 1 전자태그의 궤적 표현

RFID 시스템에서의 태그의 궤적은 그림1과 같이 전자태그가 판독기의 인식영역 안에 들어갈 때와 나갈 때 보고된 시공간 위치를 연결한 선분으로 표현된다. TPIR-Tree에서 태그의 궤적은  $tr_{tag} = \{(x, y, t) \in R^3 \mid x = x_r, y = y_r, t_{enter} \leq t \leq t_{leave}\}$  으로 표현하고 있다. 판독기 안에 들어와서 위치 보고를 하고 판독기의 인식영역 밖으로 나가지 않은 태그의 선분은 now interval이라고 정의하고, 판독기 안으로 들어와서 위치를 보고하고 판독기 밖으로 나갈 때 위치 보고를 한 태그의 선분은 fixed interval이라고 한다.

3.2 문제정의

TPIR-Tree는 RFID환경에서 전자태그 객체의 특성을 고려한 색인이다[2]. 전자태그 객체가 Enter 이벤트를 발생 시킨 후 Leave 이벤트를 보고하지 않고 머무는 태그를 표현하는 방법을 제안하였으며, 연결되지 않는 전자태그 객체의 궤적을 추적하기 위하여 태그 식별자의 차원을 추가하였다.

그러나, TPIR-Tree의 분할 정책은 R\*-Tree의 분할 기법을 그대로 사용하고 있다. 공간축에 대해서 전자태그의 정보는 점의 형태로 나타나고 이는 R\*-tree같은 시공간 색인의 사용 시 문제가 없지만, 시간축 분할시 공간 활용도와 검색 성능이 떨어진다.

TPIR-Tree의 특성상 태그 객체의 위치보고는 현재의 위치를 보고하는 것으로서 색인에서 시간 도메인 및 궤적의 공간적인 확장을 의미하며, 결과적으로 노드의 오버플로를 유발한다. 일반적인 공간 색인에서 노드들 간의 삽입 확률이 같다는 가정을 가지지만 전자태그 위치 보고의 경우에 삽입은 시간의 순서를 가지기 때문에, R\*-Tree는 낮은 공간 활용도를 가지게 된다. 즉, 전자태그의 삽입은 시간의 순서를 가지며, 과거의 노드에 새로운 데이터 삽입은 일어나지 않는다. 따라서 시간 축으로 분할 시 R\*-Tree의 균등분할 기법을 적용하면 과거의 노드에는 더 이상의 데이터의 삽입이 일어나지 않으므로 공간 활용률이 떨어진다.

4. TPIR-Tree의 시간축 분할 분석

시간축 분할의 경우에 대하여 아래의 표1과 같이 정리할 수 있다.

표 1 시간축 분할 분석

노드의 구성	두 노드간의 겹침 정도	비고
fixed interval로만 구성되어 있는 경우	포함되는 경우	데이터의 추가 삽입시 문제 발생
	겹침이 존재	문제없음
	겹침이 없는 경우	같은 마진을 가지는 축이 존재 할 수 있음
fixed / now interval이 존재하는 경우	포함되는 경우	데이터의 추가 삽입시 문제 발생
	겹침이 존재	문제없음
	겹침이 없는 경우	같은 마진을 가지는 축이 존재 할 수 있음
now interval로만 구성되어 있는 경우	포함되는 경우	데이터의 추가 삽입시 문제 발생
	겹침이 존재	발생 하지 않음
	겹침이 없는 경우	같은 마진을 가지는 축이 존재

표1에서 한 노드가 다른 노드에 포함되는 경우 R\*-Tree의 분할 축 선정 방법은 각 축에 대하여 노드의 엔트리를 lower, upper value로 정렬하여 마진을 계산한다. 시간 축에 대하여 노드안의 선분들을 시작점을 기준으로 정렬하여 마진을 계산할 때 상대적으로 짧은 간격을 가지는 선분들이 최근 시간에 몰려 있는 경우에 발생할 수 있다. 이 경우 노드가 분할이 되었지만 분할된 작은 노드가 큰 노드에 포함이 되어 이후 새로운 데이터의 삽입 시 확장영역을 비교할 때 포함되어진 작은 노드에 더 이상의 삽입이 일어나지 않는 문제가 발생 할 수 있다.

노드들 간의 겹침이 없고 시간 축에 대해서 두개의 노드로 양분되는 경우가 아닐 때에는 다른 축에서도 같은 마진 값을 가진다.

겹침이 존재하는 경우는 일반적인 경우로 이후의 데이터 삽입 시 데이터는 시간 축에서 오른쪽에 있는 노드로 삽입이 일어나게 된다. 그러나 분할될 노드가 now interval로만 구성되고 겹침이 존재할 경우는 시간축이 아닌 다른 축에 대한 마진이므로 시간축이 분할 축으로 선정되지 않는다.

5. 분할 정책

이 장에서는 노드의 저장 효율을 높이고 검색 성능의 향상을 위한 분할 축 선정기법과 시간축 비균등 분할 기법을 제시한다.

5.1 분할축 선정 기법

분할 시 한 노드가 다른 노드에 포함이 되는 경우 과도한 겹침 문제로 인하여 검색 성능이 저하되고, 이후의 데이터 삽입 시 포함되는 노드의 확장이 불가능하게 되어 저장 공간의 효율 및 검색 성능이 저하되므로, 분할 축 선정 시 포함관계의 겹침이 발생할 경우 시간축이 아닌 다른 축으로 한다.

노드 분할시 노드들 간의 겹침이 없는 경우는 두 가지 경우가 존재 한다. 첫째, 그림2의 경우처럼 시간 축으로 두개의 노드가 양분 되는 경우이다.

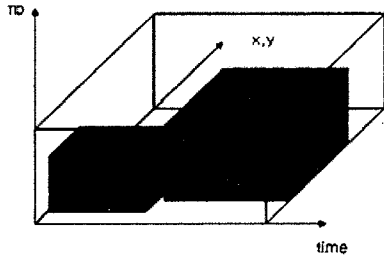


그림 2 노드 분할시 시간축으로 양분되는 경우

둘째, 그림 3처럼 노드의 엔트리들이 시간축 정렬시 최소 마진값을 가지지만 실제 두 노드는 다른축에 의해 양분되는 경우이다. 엔트리의 시작점을 기준으로 정렬할 때 발생 할 수 있다.

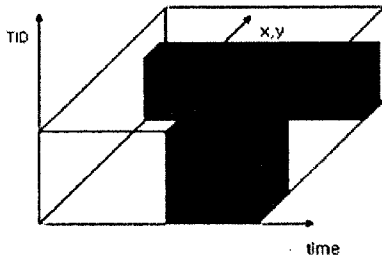


그림 3 노드 분할시 다른축으로 양분되는 경우

첫 번째 경우에 대해서는 비균등 분할 기법을 사용할 수 있지만, 두 번째의 경우에는 분할 축은 시간축이지만 실제는 다른 축으로 양분 되는 경우이므로 시간축 비균등 분할시 과도한 겹침과 사정영역이 생긴다. 이 경우 시간축에 대한 우선순위를 낮게 하여 같은 마진일 경우 다른축으로 분할을 하게 한다.

5.2 시간축 비균등 분할

시간축이 분할축으로 선정되는 경우는 한 노드가 다른 노드에 포함되는 경우와 다른축으로 양분되는 경우를 제외하고 그림4처럼 되는 경우이다.

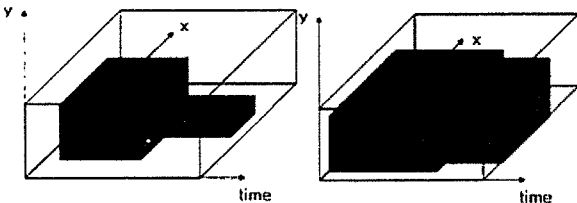


그림 4 시간축 분할이 되는 경우

시간축으로 분할할 때 두 노드를 시간축에 대하여 왼쪽 노드는 과거 노드, 오른쪽 노드는 새 노드라고 할 때, 항상 Now interval은 새 노드에 존재하게 된다. 전부 Now Interval만 있는 경우는 다른축의 마진이 더 작아서 시간축이 분할축으로 선정 되는 경우는 없다. 과거 노드에 대하여 더 이상의 삽입이 생기지 않으므로 두 노드에 대하여 비균등 분할 기법을 사용한다. 분할될 노드의 구성요소와는 관계가 없고 시간축이 분할축으로 선정 되었을 때 표2의 알고리즘을 사용하여 분할할 수 있다.

표 2 시간축 비균등 알고리즘

```

Algorithm SplitTimeAxis
STA1 For M+1 entry of node N, sorting by each
      entry's enter_time.
STA2 PN is past node, NN is new node.
      PN ← 1 to M/2 entry in N
      NN ← remainder entry in N
STA3 OS is Overlapsize. DI is new node's size
      while(NN.size > 2) {
          PN ← NN.getFront_entry();
          NN.delete_front_entry();
          IF(PN.OverlapSize(NN) < OS)
              OS ← PN.OverlapSize(NN);
          DI ← NN.size;
      }
      return DI;
    
```

6. 결론

이동체와 유사하게 태그 객체의 위치는 시간에 따라 변화하지만 전자태그 객체는 그 위치가 판독기의 위치로 정해지고 위치보고가 비주기적이며 단지 판독기의 인식영역 안에서만 위치보고가 이루어지는 특성이 있다. 또한 태그의 위치보고가 시간의 순서를 가지므로 TPIR-Tree의 균등분할시 과거노드의 저장효율이 떨어지는 문제가 발생한다.

이 논문에서는 TPIR-Tree의 분할 정책의 문제점을 소개하고 전자태그의 특성상 궤적의 보고가 완료된 선분과 완료되지 못한 선분을 가진 노드의 분할시 문제점을 분석하였다. 그리고 전자태그의 특성을 고려하여 분할시 축 선정 방법과 저장효율을 향상시킬 수 있는 비균등 분할 방식을 제시하였다. 향후 시간축 비균등 분할 알고리즘을 개선하고 기존 시공간 색인 및 TPIR-Tree와의 성능평가를 통한 검증이 필요하다.

참고문헌

- [1] K.Romér, T.Schoch, F.Mattern and T. Dubendorfer, "smart Identification Frameworks for Ubiquitous Computing applications". In Pervasive Computing and Communications Proceedings of the First IEEE International Conference, 256-262, 2003
- [2] 반재훈, 홍봉희, "RFID 시스템에서 태그의 위치 추적을 위한 시간 매개변수 간격 모델링 기법", 한국정보처리학회 춘계학술발표대회 논문집, 12권 1호, pp. 129 - 132, 2005
- [3] N. Beckmann, H. Kriegel, R. Schneider, B. Seeger, "The R\*-tree : An Efficient and Robust Access Method for Points and Rectangles", Proc. of the ACM SIGMOD International Conf. on Management of Data, pp.332-331, 1990
- [4] 전봉기, 홍봉희, "이동체의 색인을 위한 시간 기반 R-트리의 설계 및 구현", 한국정보과학회 논문지 D - 데이터베이스, 30권 3호, pp.320-335, 2003
- [5] Y. Theodoridis, M. Vassilakopoulos, and T. Sellis. "Spatio-temporal indexing for large multimedia applications", In Proc. of the 3rd IEEE Conf. on Multimedia Computing and Systmes, pp. 441- 448, 1996