R-Tree를 이용한 퍼지 인덱스

민경인*, 신예호*, 김홍기*
*충북대학교 전자계산학과
e-mail:kathy95@orgio.net,snowman@dblab.chungbuk.ac.kr
hgkim@cbucc.chungbuk.ac.kr

Adapting R-Tree to Fuzzy Indexing

Kyoung-In Min *, Yae-Ho Shin*, Hong-Ki Kim*
*Dept of Computer Science, Chung-buk National University

요약

퍼지 데이터의 일반적 특성인 불명확한 경계의 문제는 항상 명확한 데이터만을 전제로 데이터 관리를 할 수 있는 기존의 데이터베이스 시스템에서는 이를 효과적으로 저장 관리할 수 없다는 것이다. 실세계에 존재하는 많은 현상들은 항상 명확한 값들로 귀결되지 않고 불명확한 상태로 존재하는 경우가 상당하다. 따라서 데이터베이스 시스템 내에서 이와 같이 불명확한 상태를 반영하기 위한 노력의 일환으로 퍼지 데이터에 대한 표현 및 저장 관리 기법에 대한 연구가 다수 수행되었다. 그러나 기존 연구들은 주로 데이터의 상태변화가 거의 없는 정적 환경에 적합할 뿐 값의 갱신이 빈번히 발생하는 동적 환경에는 적합하지 않은 문제가 있다. 이에 본 논문에서는 데이터 갱신이 빈번히 발생하는 동적 환경하에서 경계가 불명확한 퍼지 데이터의 관리를 효과적으로 수행하도록 하기 위한 방안으로서 R-Tree를 이용한 퍼지 데이터 색인 방법을 제안한다.

1. 서론

실세계에서, 데이터 베이스에 저장, 검색하는 데이터의 형태가 명확하다는 가정은 지리정보 시스템, 멀티미디어 시스템, 캐드(CAD),전문가 데이터베이스 시스템(Expert Database system)과 같은 복잡한 응용 프로그램들에서는 제한적이다. 이런 응용 프로그램들은 불명확한 형태로 데이터베이스에 적용되는 경우가 간혹 있다.

데이터베이스 영역에서 많은 양의 연구가 불확실성을 나타내는 데에 초점을 맞춰왔다. 그러나 비록 데이터베이스에서 응답시간이 중요한 고려사항이기는 하나 이런 결과들의 구현 측면은 충분한 주목을 받지 못하고 있다. 게다가 불확실성을 다루는 것이 저장 용량의 증가 제안된 방법의 고도의 수치적 특성 및 가장 중요한 기존의 접근(access)구조에의 부적합성과 같은 것으로 인하여 성능 저하의 부가적인 원인이 될 때는 더욱 결정적이 된다. 결과적으로, 불명확한 형태의 자료를 효율적으로 나타낼 수 있는 방법이 보다 더 필요하다[1,2].

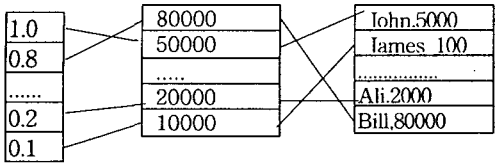
이와 관련하여 기존의 방법에는 Yazici[1]와 Bosc[2]가 제시한 인덱스 방법이 있으나 이들 방법은 정적인 환경에 적합한 방법이다. 따라서, 자료의 삽입과 삭제, 갱신이 빈번하게 발생하는 동적인 환경에서는 부적합하다.

그러므로 본 논문에서는 동적인 환경 하에서 R-Tree[4]를 이용하여 효율적으로 불확실한 데이터를 나타내고 접근하는 방법을 제안 하고자 한다. 이를 위해, 불확실한 데이터를 역 퍼지화(defuzzification) 시킨 후 R-Tree[4]를 적용한다. 2장에서는 기존의 퍼지 인덱스에 해당하는 관련 연구를 살펴보고, 3장에서는 역 퍼지화(defuzzification) 시키는 과정과, R-Tree[4]를 적용하는 과정을 기술한다. 4장에서는 결론을 제시한다.

2. 관련연구

퍼지 인덱스를 최초로 제시한 사람은 Bosc이다. Bosc가 제시한 인덱스의 원리는 한 속성에 묶인 퍼지 속성당 하나의 인덱스를 사용하는 것이다. 그 원리는 술어를 만족하는 튜플 들의 리스트에 퍼지 술어의 구성 값(membership grade)을 대응시키는

것이다. 이 방법은 같은 도메인(Homogeneous)에서만 사용되고 기본 관계는 명확(crisp) 하다는 것을 가정한다. [그림1]에 Bosc의 인덱스 구조가 나타내어지고 있다[2].

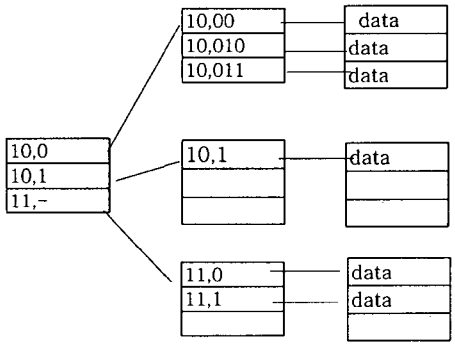


[그림 1]Bosc 인덱스 구조

[그림 1]의 Bosc가 제시한 방법은 술어당 소속값(membership grade)을 대응시키는 것이기 때문에, 이질적(heterogeneous)인 자료를 효율적으로 접속하는데 추가적인 복잡성을 일으키게 된다.

Bosc[2]이후에 제시된 색인 구조는 Yazici가 제시한 MLGF나 K-d Tree를 이용한 색인 구조가 있다. MLGF[3]를 가지고 구현한 색인 구조는 퍼지 속성(fuzzy attribute)을 불확실한 값을 가질 때와 명확(crisp)한 값을 가질 때를 두 가지를 다 고려하여, 레코드를 나타내는 방법을 달리하고 조직속성(organizing attribute)을 임의로 추출해서, 나타내어진 레코드에서, 몇 개의 비트를 뽑아 MLGF를 이용해 저장하고 접근하는 구조이다.

이질적인 도메인을 처리할 수 있는 장점과 함께 기존의 그리드 파일(Grid File)을 이용할 때보다 검색할 때 자료의 입출력 양을 많이 줄었다는 장점을 가지고 있다. MLGF를 이용한 인덱스의 구조는 다음과 같다[3].



[그림2]MLGF를 이용한 인덱스 구조

한편 B-Tree와 유사한 높이 균형트리인 R-Tree는 데이터의 갱신이 빈번히 발생하는 동적 환경에서 다차원 데이터에 대한 색인 방법으로서 가장 적합한 구조로 알려져 있다[4]. 반면 다차원 색인 구조를 갖는 MLGF의 경우 삽입, 삭제가 빈번히 발생하는 환경에서는 적당하지 못하다. 이에 본 논문에서는 삽입, 삭제가 빈번히 발생하는 동적 환경에서 불확실한 데이터를 효과적으로 색인하기 위하여 R-Tree를 이용한다.

3.퍼지 데이터의 특성

본 논문에서 적용 하고자 하는 퍼지 데이터는 불확실한 퍼지 값을 갖거나 또는 명확한 값을 표현할 수 있는 퍼지 속성을 포함한다. 아울러 퍼지 데이터로 관계형 데이터 베이스를 통해 저장 관리 되는 퍼지 속성은 기본키(primary key)가 될 수 없다.

퍼지 속성(fuzzy attribute)은 특성상 불 확실한 형태와 명확한 형태의 자료를 동시에 포함할 수 있다. 예를 들어 퍼지 속성(fuzzy attribute)이 키(height)라면 그 속성은 182㎝인 명확한 값과 크다(tall),작다(short)와 같은 불명확한 속성 값(fuzzy term)을 가질 수 있다. 이때 퍼지 속성(fuzzy attribute)에 값의 형태가 크다(tall) 작다(short)와 같이 불확실한 형태일 때에는 소속 함수(memgership function)를 이용하여 수치 도메인으로 사상이 가능하다. 숫자로 사상된 퍼지 값은 단일한 값이 아닌 일정 구간 값의 형태로 사상이 되게 된다. 하나의 퍼지 값은 동일한 구간으로 사상되므로 동일 구간에서 많은 중복이 발생할 수도 있다. 이와 같은 문제를 해결하기 위해, 퍼지 속성 값을 소속 함수(membership function)를 사용하여 수치 도메인으로 사상한 후 중복을 회피할 수 있도록 변환하는 과정이 필요하다. 사상된 구간의 변환 과정이 필요하다.

4 R-Tree를 이용한 퍼지 데이터 색인

이 장에서는 R-Tree를 이용한 퍼지 데이터 색인을 설명하기 위해 키 데이터에 포함될 퍼지 속성의 변환과정을 설명한다.

4.1 퍼지 속성값의 변환

앞 장에서 불 명확한 퍼지 속성 값을 소속 함수(membership function)를 이용하여 수치 도메인으로 사상할 수 있음을 고찰하였다. 이와 같이 퍼지 소속 값을 수치 도메인으로 사상하는 과정을 역퍼지화(defuzzification)라고 한다. 본 논문에서 예제로써 다루는 키 데이터의 퍼지 속성을 역 퍼지화(defuzzification) 시키기 위하여, α-cut의 개념을 이용한다. 역 퍼지화(defuzzification) 과정을 설명하기 위한 기본 레코드 구조는 다음과 같다.

```
record Employee
  string Name
  integer SID
  fuzzy Height
end
```

이 레코드구조에 따라 퍼지 데이터인 키(height)에 적용하는 소속함수(membership function)는 아래와 같다.

Short:1,35~1,70

$$\text{short}(x)=\begin{cases}1 & x<1.35\\ c(\alpha-x)^2 & 1.35\le x<\alpha\\ 0 & x\ge\alpha\end{cases}$$

where $\alpha = 1.7$ $c=1/0.1225$

Medium:1,57~1,85 (c=22.222)

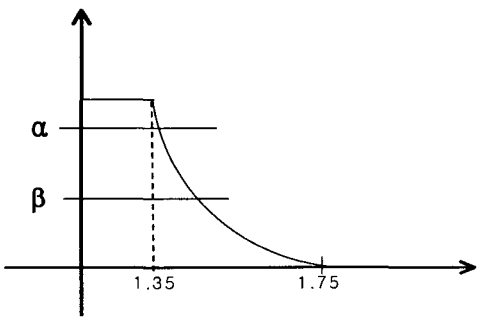
$$\text{Medium}(x)=\begin{cases}0 & x<1.4\\ c(1.4-x)^2 & 1.4\le x<1.55\\ 1-c(x-1.7)^2 & 1.55\le x<1.85\\ c(2-x)^2 & 1.85\le x<2.00\\ 0 & x\ge 2.00\end{cases}$$

Tall:1,86-2,00

$$\text{Tall}(x)=\begin{cases}0 & x\le\alpha\\ c(x-\alpha)^2 & \alpha\le x<2.00\\ 1 & x\ge\end{cases}$$

where $c=1/0.1996$ $\alpha = 1.86$

위에서 제시된 함수에 α-cut의 개념을 넣어 역퍼지화(defuzzification)하게 된다. 이때 α의 값은 0.8로 하고 자료의 값이 short일 경우에 다음과 같은 그래프가 만들어 지게된다. [그림 3] 이 때 α의 값을 대입한 후 x의 값을 구하면 된다. 역 퍼지화(defuzzification)후의 x의 값은 항상 같은 구간으로 나오게 된다. 따라서 항상 동일한 색인 영역을 점유하게 되어, 색인 내에서 지나친 중복 현상을 유발시킨다. 이와 같은 문제를 해결하기 위하여 여기서 β값 0.5로 다시 한 번 역 퍼지화(defuzzification)하여 중첩을 피하기 위한 구간을 구하는 것이다.



[그림 3] short 함수의 소속 함수

4.2 퍼지데이터의 적용을 위한 R-Tree 알고리즘

이 절에서는 불확실한 퍼지 데이터를 R-Tree에 적용할 수 있도록 하기 위하여 4.1절에서 언급한 역퍼지화 과정(defuzzificaiton)과 구간 분산에 대한 알고리즘을 R-Tree 알고리즘에 확장 적용한다.

<역퍼지화 알고리즘>

I1. $x = f^{-}(y)*\alpha$ (y = 퍼지 속성)

<구간 분산 알고리즘>

```
I1. D  = SID% I(I=f⁻(y)*β)
    while(D>10)
    {
     D  = SID% I(I=f⁻(y)*β)
    }
  if(SID1 < SID2 ) x=x+ D
  else x =x-D
```

<삽입 알고리즘>

I1.if height = 불확실한 값 역퍼지화 알고리즘과 분산 알고리즘을 호출한다.
I2.새로운 레코드에 대한 위치를 찾는다.
I3.단말 노드에 레코드를 추가한다.(만일,빈자리가 없으면 분할 한다.)
I4.트리의 상향 전파

여기서 I1은 퍼지 속성 값을 R-Tree에 적용하기 위하여 역 퍼지화(defuzzification) 및 분산 알고리즘을 호출하는 부분이고 I2~I4는 R-Tree의 삽입 알고리즘이다.

4.3 사례연구

이 절에서는 지금까지 기술한 퍼지 데이터 색인 방법에 표[1]의 예제 데이터를 적용하여 전체적인 동작과정을 시뮬레이션 한다. 표[2]는 표[1]의 원시 데이터에 대하여, 역 퍼지화(defuzzification)과정을 통해 생성된 자료를 나타낸다.

[표1] 원본 자료

번호	이름	SID	Height
1	Kirk Brown	91858	172
2	Terry Stallon	92617	short
3	Billy Brown	99049	175
4	Sally Fields	93183	155
5	SusanJacksn	91927	tall
6	SharonGingrich	92899	152
7	Bruce Oldhouse	95450	181
8	Jefferson Abel	92244	168
9	Michael Brown	99257	tall
10	Bill Lincoln	90735	173
11	Michael Idol	99642	183

[표2] 키의 불확실한 자료를 역퍼지화한 값

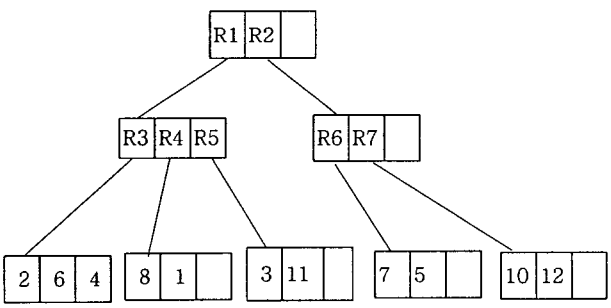
번호	SID	Hegiht	역 퍼지화한 값
1	91858	172	
2	92617	short	135~138
3	99049	175	
4	93183	155	
5	91927	tall	186~195
6	92899	152	
7	95450	181	
8	92244	168	
9	99257	tall	188~197
10	90735	173	
11	99642	183	

여기서,5번 레코드와 9번 레코드가 역 퍼지화(defuzzification)한 구간 값이 같으므로, 4.2에서 제시한 <구간 분산 알고리즘>을 5번 레코드와 9번 레코드의 경우에 적용하면, 다음과 같다.

SID:99257
I: 81
D= |99257%81| = ± 32

32는 범위를 벗어나므로, 다시 한 번 나눠 준다. 이 때 나오는 나머지는 ±2가된다. ±2 중에서, 앞에 있는 SID 값이 더 작으므로 +값을 취해주었다. 따라서 9번 레코드의 역 퍼지화(defuzzification)된 값은 188~197이 되게 된다. 퍼지 속성 값에서 변형된 구간은 2차원 공간 상에서 (a,b) 두 점으로 나타내어질 수 있다. 또한, 명확한 형태의 자료는 2차원 공간 상에서 (c,c)이런 형태로 나타 내어진다. 위에서 기술된 변형된 데이터의 형태로, R-Tree의 연산이 적용 되어진다.

[표2]에 있는 자료를 바탕으로 R-Tree를 구성해 보면 다음과 같다. [그림 4]



[그림4][표2]의 자료가 삽입된 트리

5.결론

본 논문에서는 동적인 환경 하에서 불확실한 특성을 갖는 퍼지 데이터를 효율적으로 관리하기 위한 방법으로써 R-Tree를 사용하는 방법을 제시하였다. 이는 퍼지 속성(fuzzy attribute)의 불확실한 값(fuzzy term)에 대해 소속 함수를 이용한 역 퍼지화(defuzzify)를 통해 특정 구간으로 사상하는 과정을 포함한다. 특정 구간으로 사상된 퍼지 속석값은 다시 일정 구간 내로 수렴시키는 분산을 통해 R-Tree에 적용할 수 있는 데이터로 변환시키고 변환된 데이터를 R-Tree에 적용할 수 있도록 하였다. 아울러 이 논문에서 제안하고 있는 방법에 대한 사례 연구를 통해 제안 방법이 동작하는 전 과정을 검증하였다.

향후 연구 과제로는 본 논문에서 제안한 방법에 대한 구현 및 실험을 통해 본 논문에서 제안한 방법에 대한 타당성을 검증하는 것이 필요하다.

[참고문헌]

[1]Yazici "Index Structure for Fuzzy Database" IEEE,1996
[2]Bosc P.Indexing Principles for a Fuzzy Database",Information systems,1989
[3]K.Y.Whang, Multi-level Grid File-A Dynamic Hierarchical Multidimensional File Structure,"Database Systemes for Advanced Applications 1994
[4]Autonin Guttman,R-Trees:A Dynamic Index Structure For Spatial Searching ACM 1984
[5]Zadeh L.A.,Similarity Relations and Fuzzy Orderings,"Information Sciences,1971
[6]George J.Klir,Tina A.Folger,Fuzzy sets, Uncertainty,And Information, Prentice hall,
[7]Zadah,LA, "FuzzySets",Information and Control,Academic Press,1965