

차원 분리 기법을 적용한 3차원 공간 위상 관계 연산자 설계*

전성우⁰, 지정희, 김상호, 류근호
충북대학교 데이터베이스 연구실

e-mail : {swjun⁰, jhchi, shkim, khryu}@dmlab.chungbuk.ac.kr

Design of 3D Spatial Topological Relationships Using The Dimension Separated Method

Sung Woo Jun⁰, Jeong Hee Chi, Sang Ho Kim, Keun Ho Ryu
Database Laboratory, Chungbuk National university

요 약

기존 연구에서 3차원 공간 위상 관계 연산자를 정의하는 방법은 2차원 공간 위상 관계 정의를 위해 주로 사용되는 공간 교차 방법을 단순히 3차원으로 확장해서 정의하는 방식이다. 그러나 기존의 방법에 의해 정의된 3차원 위상 관계 연산자로는 3차원 데이터베이스를 이용한 실제 응용 시스템에서 고려되어야 하는 특수한 위상 관계 분석을 수행하기 어렵다. 따라서 이 논문에서는 실제 응용에서 적용 될 수 있는 특수한 3차원 위상 관계를 정의 하기 위하여 x,y 좌표와 z좌표를 분리해서 고려하는 새로운 기법을 제안한다. 또한 제안한 방법을 이용하여 새로운 3차원 공간 위상 관계 연산자들을 정의하고 구체적인 알고리즘을 제시한다.

1. 서론

최근 컴퓨팅 기술의 발달에 따라 대용량 데이터베이스의 처리가 가능해지면서 지리 정보 시스템이나 공간 데이터베이스 분야에서 3 차원 공간데이터 베이스에 관한 연구가 활발히 진행되고 있다. 특히 공간 데이터베이스에서 가장 중요한 부분이라고 할 수 있는 인덱스와 공간 위상 관계에 관한 연구가 많은 연구자들에 의해서 수행되었다.

그러나 기존의 3 차원 공간 위상 관계 연산자들은 단순히 차원만 확장하여 설계 되었기 때문에 3 차원 공간 객체들 사이에서 가능한 특수한 위상 관계를 표현하는데 적용할 수가 없다. 예를 들어 어떤 도로와 그 도로 위를 지나가는 고가 도로 사이의 관계 또는 어떤 건물의 아래에 위치하는 가스 배관 사이의 관계를 기존의 3 차원 위상 관계 연산자로 분석할 경우 서로 떨어져 있는 것으로 표현된다. 그러나 기존의 2차원 위상 관계 연산자로 분석할

경우 서로 교차하는 것으로 분석된다. 즉 3 차원 상에서는 서로 교차하지 않지만 2 차원 상에서는 교차하는 경우를 분석하기 어렵다.

따라서 이 논문에서는 이러한 문제점을 해결하기 위하여 새로운 3 차원 위상 관계 정의 기법을 제안하고 제안된 방법에 따라 3 차원 공간 위상 관계 연산자들을 정의하고 구체적인 알고리즘을 제시한다.

이 논문의 전개를 위하여 제 2 장에서는 기존의 2 차원 및 3 차원 공간 위상 관계 정의에 대해 기술한다. 제 3 장에서는 새로운 3 차원 공간 위상 관계 정의 방법을 제안하고, 제안된 방법에 따라 3 차원 공간 위상 관계 연산자를 정의한다. 제 4 장에서는 정의한 연산자의 구체적인 알고리즘을 기술하고 마지막으로 제 5 장에서 결론을 맺는다.

2. 관련 연구

이 장에서는 관련연구로서 기존의 위상관계 정의 방법과 3 차원 공간 위상 관계 정의 방법에 대해

* 이 연구는 한국과학재단 지원 RRC(청주대 ICRC)의 지원으로 수행되었음.

기술하고 문제점을 제시한다.

2.1 2차원 공간 위상관계 정의 방법

2 차원 위상 관계를 정의하는 방식은 크게 포인트 집합에 기반 한 공간 교차 방법[1]과 계산에 기반 한 방법(Calculus based Method)[2]으로 분류된다. 이 중에서 널리 인정된 방법은 공간 교차 방법이다[3]. 공간 교차 방법에는 4-Intersection Method(4IM)[4], 9-Intersection Method(9IM)[5], Dimension extended Method(DEM)[6], DE-9IM[2]등이 있다.

공간 교차 방법은 공통적으로 포인트 집합(point set)을 기반으로 한다. 4-IM 에서는 객체들의 내부와 외부 사이의 교집합 관계의 조합으로 위상관계를 표현하고, 9IM 에서는 객체의 내부, 경계, 외부를 나타내는 $I(a)$, $B(a)$, $E(a)$ 사이의 교집합 결과들의 조합으로 조건을 공식화 한다. DEM 은 4IM 을 확장한 것으로 4IM 의 교집합 결과에 대한 차원 값 -1, 0, 1, 2 의 조합으로 위상 관계를 나타낸다. 마지막으로 DE-9IM 은 9IM 을 차원으로 확장한 것으로 객체의 내부, 경계, 외부 사이의 교집합 결과와 교집합 결과 기하가 가질 수 있는 차원 값(-1,0,1,2)의 조합을 8 개의 패턴 값으로 나타내어 공간 관계 조건을 공식화 하는 것이다. 표 1 은 DE-9IM 패턴 행렬을 나타내고 있다.

	I(내부)	B(경계)	E(외부)
I(내부)	$dim(I(a) \cap I(b))$	$dim(I(a) \cap B(b))$	$dim(I(a) \cap E(b))$
B(경계)	$dim(B(a) \cap I(b))$	$dim(B(a) \cap B(b))$	$dim(B(a) \cap E(b))$
E(외부)	$dim(E(a) \cap I(b))$	$dim(E(a) \cap B(b))$	$dim(E(a) \cap E(b))$

표 1. DE-9IM 패턴 행렬

Egenhofer[1]는 9IM 을 이용하여 disjoint, meet, overlap, equal, covers, coveredBy, inside, contains 등의 8 가지 2 차원 공간 위상 관계 연산자를 정의 하였고, OpenGIS 컨소시엄[7]에서는 DE-9IM 을 이용하여 Disjoint, Equal, Intersect, Touch, Cross, Within, Contain, Overlap 등의 7 가지 2 차원 공간 위상 관계 연산자를 정의하였다.

2.2 3차원 공간 위상 관계 정의 방법

3 차원 공간 위상 관계에 관한 연구가 몇몇 연구자들에 의해 수행 되었다[8,9,10]. 그러나 그들은 2 차원 위상 관계를 단순히 3 차원으로 확장하여

가능한 관계들을 제시하였다. [8]에서는 DE-9IM 을 3 차원으로 확장한 3DE-9IM 을 제안하고 이 방법을 사용하여 3 차원 위상 관계 연산자들을 정의하였다. [9]에서는 기존의 여러 가지 2 차원 위상 관계 정의 기법을 적용하여 3 차원 공간 위상 관계를 정의하였다. [10]에서는 9IM 을 이용하여 0, 1, 2 차원 그리고 3 차원 공간에서 다차원 객체들 사이의 가능한 위상 관계에 관한 통합된 연구를 보여주었다.

그러나 지금까지 기술한 기존의 연구에서는 3 차원 객체들 사이에서 가능한 특수한 위상 관계는 제시되지 않았다. 따라서 이러한 기존의 연산자를 사용하면 2 차원 공간데이터베이스로 표현할 경우에 중요하게 간주되었던 몇몇 공간 위상 관계(예를 들면 건물 아래로 가스관이 지나가는 경우)들이 무시되는 결과를 가져온다. 따라서 실제 응용에 적용될 수 있는 3 차원 위상 관계를 정의하기 위한 새로운 정의 기법이 필요하다.

3. 새로운 3차원 공간 위상 관계 정의 기법

이 논문에서 제안하는 3 차원 공간 위상 관계 정의 기법은 3 차원 객체의 경계(B), 내부(I), 외부(E) 사이의 교집합 관계를 이용하는 9IM 을 적용할 때 x, y 좌표와 z 좌표를 구분하여 사용하는 방식이다. 즉, 3 차원 객체의 x,y 좌표상의 경계($B_{x,y}$), 내부($I_{x,y}$), 외부($E_{x,y}$)들 사이의 교집합 관계를 이용 하고 객체의 가장 높은 z 좌표 z_{top} 과 가장 낮은 z 좌표 z_{bottom} 사이의 포함 관계를 적용한다. 이 논문에서는 이 방식을 차원 분리 교집합 접근법(Dimension Separated - 9 Intersection Method, DS-9IM) 이라고 명명한다. 이 논문에서는 DS-9IM 기법을 이용하여 8 개의 기본적인 3 차원 공간 관계 연산자와 한 객체가 다른 객체의 아래쪽에 위치한 경우의 7 가지 3 차원 공간 관계 연산자, 그리고 한 객체가 다른 객체의 위쪽에 위치한 경우의 7 가지 3 차원 공간 관계 연산자들을 정의한다. 첫 번째의 경우는 3Dxxx 계열 연산자로 명명하고 두 번째와 세 번째의 경우는 각각 3DLxxx 와 3DUxxx 계열 연산자로 명명한다. 여기서 L 과 U 는 각각 Lower 와 Upper 의 약자이다. 제안하는 기법으로 3 차원 공간 위상 관계 연산자를 정의하면 다음과 같다.

[정의 1] 3Dinside 연산은 x, y, z 좌표상에서 한 객체가 다른 객체의 내부에 위치하는 경우를 의미하는 것으로 다음 식으로 정의한다.

$$a.3Dinside(b) \Leftrightarrow (I_{x,y}(a) \cap I_{x,y}(b) \neq \emptyset) \wedge (E_{x,y}(a) \cap I_{x,y}(b) \neq \emptyset) \wedge (B_{x,y}(a) \cap B_{x,y}(b) = \emptyset)$$

$$\wedge (z_{top}(a) > z_{bottom}(b)) \wedge (z_{bottom}(a) < z_{top}(b))$$

[정의 2] 3DLinside 연산은 한 객체가 다른 객체의 아래에 위치하면서 z 좌표 상으로는 떨어져 있고 x, y 좌표상으로는 내부에 위치하고 있는 경우를 의미하는 것으로 다음 식으로 정의한다.

$$a.3DLinside(b) \Leftrightarrow (I_{x,y}(a) \cap I_{x,y}(b) \neq \emptyset) \wedge (E_{x,y}(a) \cap I_{x,y}(b) \neq \emptyset) \wedge (B_{x,y}(a) \cap B_{x,y}(b) = \emptyset) \wedge (z_{top}(a) < z_{bottom}(b))$$

[정의 3] 3DUinside 연산은 한 객체가 다른 객체의 위에 위치하면서 z 좌표 상으로는 떨어져 있고 x, y 좌표상으로는 내부에 위치하고 있는 경우를 의미하는 것으로 다음 식으로 정의한다.

$$a.3DUinside(b) \Leftrightarrow (I_{x,y}(a) \cap I_{x,y}(b) \neq \emptyset) \wedge (E_{x,y}(a) \cap I_{x,y}(b) \neq \emptyset) \wedge (B_{x,y}(a) \cap B_{x,y}(b) = \emptyset) \wedge (z_{bottom}(a) > z_{top}(b))$$

그림 1은 제안한 22 개의 3 차원 공간 위상 관계를 가지적으로 나타낸 것이다. 이 그림에서는 3Dinside/3Dcontain 와 같이 연산 대상의 순서만 다르고 위상 관계는 유사한 것은 둘 중에서 하나만 나타내었다. 또한 3DUxxx 계열 연산자들은 3DLxxx 계열 연산자들과 위/아래 순서만 바꾸면 관계가 동일하므로 그림으로 표기하지 않았다. 그림 1 에서 a)부터 f)까지 3Dxxx 계열 연산자들은 기본적인 3 차원 공간 위상 관계를 보여주고 있고, g)부터 k)까지 3DLxxx 계열 연산 자들은 한 객체가 다른 객체의 아래쪽에 위치한 경우를 보여주고 있다.

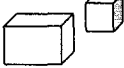
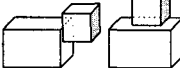




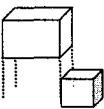
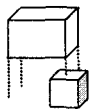
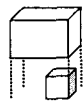
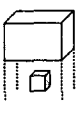
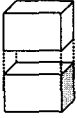
		
a) 3Ddisjoint	b) 3Dmeet	c) 3Doverlap
		
d) 3Dcover	e) 3Dinside	f) 3Dequal
		
g) 3DLmeet	h) 3DLoverlap	i) 3DLcover
		
j) 3DLinside	k) 3DLequal	

그림 1.3 차원 부피 객체들 사이의 공간 위상 관계 이와 같이 정의한 22 개의 기본적인 3 차원 공간 위상 관계 중에서 3Dmeet 와 3Ddisjoint 연산은 x,y 좌표와 z 좌표의 기본적인 분리 기법과 다른 적용이 필요한 것으로 다음과 같이 정의 하였다.

[정의 4] 3Dmeet 는 그림 1 의 c)와 같이 객체와 객체가 직접 만나는 상태를 의미하는 것으로 x,y 좌표와 z 좌표를 분리하지 않고 정의한다. 따라서 정의는 다음과 같다.

$$a.3Dmeet(b) \Leftrightarrow (I_{x,y,z}(a) \cap I_{x,y,z}(b) = \emptyset) \wedge ((I_{x,y,z}(a) \cap B_{x,y,z}(b) \neq \emptyset) \vee (B_{x,y,z}(a) \cap I_{x,y,z}(b) \neq \emptyset) \vee (B_{x,y,z}(a) \cap B_{x,y,z}(b) \neq \emptyset))$$

[정의 5] 3Ddisjoint 는 z 좌표와 관계없이 x,y 좌표를 기준으로 서로 떨어져 있는 경우를 의미하는 것으로 다음 식으로 정의한다.

$$a.3Ddisjoint(b) \Leftrightarrow (I_{x,y}(a) \cap I_{x,y}(b) = \emptyset) \wedge (I_{x,y}(a) \cap B_{x,y}(b) = \emptyset) \wedge (B_{x,y}(a) \cap I_{x,y}(b) = \emptyset) \wedge (B_{x,y}(a) \cap B_{x,y}(b) = \emptyset)$$

지금까지 이 논문에서 제안한 DS-91M 기법에 따라 3 차원 부피 객체들 사이의 위상 관계를 정의하였다. 점, 선, 면 등 다른 타입의 3 차원 객체들 사이의 위상 관계도 위와 같이 정의 된다. 그러나 몇몇 객체들 사이에서는 정의될 수 없는 예외 상황이 존재한다. 예를 들면 두 점 객체들 사이에서는 3Doverlap 관계가 성립될 수 없다. 이에 관한 자세한 사항은 [7,9]에 상세히 기술되어 있다.

4. 3 차원 위상 관계 연산자 알고리즘

이 장에서는 향후 구현을 위하여 다음과 같이 제안한 연산자들의 구체적인 알고리즘을 기술하였다.

[알고리즘 1] 3Doverlap 연산

```

Algorithm 3Dinside(3DsourceObject, 3DtargetObject)
Input : 3D Geometry 타입의 공간객체
Output : Boolean(true or false)
Boolean 3Dinside(3DsourceObject, 3DtargetObject)
{
  get Oid of 3DsourceObject and 3DtargetObject }
  If((ztop(3DsourceObject) > zbottom(3DtargetObject)) ∧
  (zbottom(3DsourceObject) < ztop(3DtargetObject))) {
    두 객체의 x,y 좌표를 검색해서 배열에 저장;
    If(x,y 좌표상에서 저장된 두 객체의 inside가 성립)
      return true;
    Else
      return false;
  }
  Else
    return false;
}
    
```

[알고리즘 2] 3DLinside 연산

```

Algorithm 3DLinside(3DsourceObject, 3DtargetObject)
Input : 3D Geometry 타입의 공간객체
Output : Boolean(true or false)
Boolean 3DLinside(3DsourceObject, 3DtargetObject)
{ get Oid of 3DsourceObject and 3DtargetObject }
  If( $z_{top}(3DsourceObject) < z_{bottom}(3DtargetObject)$ ){
    두 객체의 x,y 좌표를 검색해서 배열에 저장;
    If(x,y 좌표상에서 저장된 두 객체의 inside가 성립)
      return true;
  }
  Else
    return false;
}
Else
  return false;
}
    
```

```

Algorithm 3DUinside(3DsourceObject, 3DtargetObject)
Input : 3D Geometry 타입의 공간객체
Output : Boolean(true or false)
Boolean 3DUinside(3DsourceObject, 3DtargetObject)
{ get Oid of 3DsourceObject and 3DtargetObject }
  If( $z_{bottom}(3DsourceObject) > z_{top}(3DtargetObject)$ ){
    두 객체의 x,y 좌표를 검색해서 배열에 저장;
    If(x,y 좌표상에서 저장된 두 객체의 inside가 성립)
      return true;
  }
  Else
    return false;
}
Else
  return false;
}
    
```

[알고리즘 3] 3DUinside 연산

알고리즘 1, 알고리즘 2 그리고 알고리즘 3 은 각각 3Dinside 연산, 3DLinside 연산, 3DUinside 연산을 나타내고 있다. 제시된 알고리즘들은 공통적으로 먼저 z 좌표들 사이의 관계를 검사한 후 조건을 만족하면 x,y 좌표상에서 공간 위상 관계의 성립 여부를 검사한다. 제시된 알고리즘들의 차이점은 z 좌표의 조건이다.

5. 결론 및 향후 연구

기존의 3 차원 위상 관계 정의 방식으로는 3 차원 공간 객체들 사이에 나타날 수 있는 특수한 위상 관계들을 정의할 수 없기 때문에 기존의 방식으로 정의된 3 차원 공간 위상 관계 연산자를 이용하면 실제 응용에서 중요하게 다루어져야 할 몇몇 위상 관계들을 분석할 수가 없다. 이러한 문제를 해결하기 위해 이 논문에서는 새로운 3 차원 위상 관계 정의 방법인 차원 분리 교집합 기법을 제안하였다. 또한 제안한 기법으로 22 개의 3 차원 공간 위상 관계

연산자들을 정의하였고 구체적인 알고리즘을 제시하였다.

향후 연구로는 제안한 연산자들을 구현하여 실제 응용 시스템에 적용해 보고 평가하는 작업을 수행할 예정이다.

참고문헌

- [1] M.J. Egenhofer and R.D. Franzosa, "Point-set Topological Spatial Relations," International Journal of Geographical Information Systems, Vol. 5, pp. 161-174, 1999.
- [2] E. Clementini and P. Di Felice, A Comparison of Methods for Representing Topological Relationships, Information Sciences, pp. 149-178, 1995.
- [3] M.Erwig and M.Schneider, "Developments in Spatio-Temporal Query Languages," In IEEE Int. Workshop on Spatio-Temporal Data Models and Languages, 1999.
- [5] M.J. Egenhofer and John R. Herring, "Categorizing binary topological relationships between regions, lines, and points in geographic databases," Technical report, Department of Surveying Engineering, University of Maine, Orono, Me, 1991.
- [4] D.V. Pullar and M.J. Egenhofer, "Toward formal definitions of topological relations among spatial objects," In Proceeding of the 3rd International Symposium on Spatial Data Handling, Sydney, Australia, pp. 225-241, Columbus, OH, August 1998.
- [6] E. Clementini, P. Di Felice, and P. van Oosterom, "A small set of formal topological relationships suitable for end-user interaction," In D. Abel and B. C. Ooi, editors, Third International Symposium on Large Spatial Databases, Lecture Notes in Computer Science no.692, pp. 277-295, Springer-Verlag, Singapore, June 1993.
- [7] Open GIS Consortium, Inc. OpenGIS, "Simple Features Specification For SQL Revision 1.1," OpenGIS Project Document 99-049, 1999.
- [8] 김상호, 강구, 류근호, "3 차원 공간 위상 관계 연산자의 설계", 한국정보처리학회 논문지 D, Vol.10, No. 02, pp. 211-220, 2003년 4월.
- [9] 이성호, 김경호, 김성수, 김경욱, "3 차원 공간 객체들의 위상 관계", 한국 정보처리학회 추계 학술발표논문집, 제 9 권, 제 2 호. 2002년 11월.
- [10] S. Zlatanova, "On 3D Topological Relationships," 11th International Workshop on Database and Expert Systems Applications(DEXA'00), pp. 913, Sept 2000.