

컴포넌트 기반의 2차원 시공간 위상관계 연산자의 설계[†]

강구^{0*}, 이현아^{*}, 김상호^{*}, 류근호^{*}, 이성호^{**}

충북대학교 데이터베이스연구실

^{*}{kangih,halee,shkim,khryu}@dmlab.chungbuk.ac.kr, ^{**}shlee@etri.re.kr

A Design of the Component based 2D SpatioTemporal Topological operators

Goo Kang^{0*}, HyunAh Lee^{*}, SangHo Kim^{*}, KuenHo Ryu^{*}, SungHo Lee^{**}

^{*}Dept of Computer Science, Chung-Buk National University

^{**}ETRI

요약

2차원 공간데이터를 시간차원으로 확장한 데이터모델은 최상위의 TGeometry 컴포넌트를 갖는다. TGeometry 컴포넌트는 시공간에 관한 연산뿐만 아니라 다양한 연산들을 제공한다. 2차원 시공간 데이터모델에 대한 시공간 연산자를 설계하기 위하여, 공간에 대한 방법으로서, 차원으로 확장된 9IM 방법을 이용하였는데, 이것은 공간 객체를 내부, 경계, 그리고 외부로 구분하고, 이것들간의 관계를 차원으로 나타내는 방법이고, 또한 시간에 대한 방법으로서, OpenGIS에서 제안한 8개의 연산들을 시간에 적용했으며, 이들 두 가지 방법을 통합하여 시공간 객체에 대한 위상관계 tContains, tWithin, tOverlaps, tIntersects, tTouches, tEquals, tDisjoint, tCrosses 연산을 설계하였으며, 실제 예를 통해 적용하였다.

2. 관련연구

2.1. 2차원 시공간 객체

OpenGIS 명세서에서는 2차원 공간 객체 모델을 Geometry, Point, Curve, LineString, Polygon 등의 단일 공간 객체와 단일 객체의 집합(Collection)형태인 MultiPoint, MultiLineString, 그리고 MultiPolygon으로 나타내고, 이들 공간 객체 사이의 연산들을 제시하고 있다[OGC99]. Geometry는 공간 데이터 계층에서 루트 클래스이며 추상화된 클래스이다. 또한 시간데이터베이스에서는 유효시간(valid time), 거래시간(transaction time), 그리고 사용자 정의시간 등을 사용할 수 있다. [류근호00]에서는 OpenGIS의 2차원 공간 모델에 ValidTime이라는 추상클래스를 정의함으로써 공간모델을 시간차원으로 확장했다. 각 공간 속성은 해당 타임 스탬프를 갖고, 해당 객체의 비전들 간의 관계를 정의함으로써 이력객체와 이동객체를 표현한다. Point, Line에 타임스탬프를 함으로써 이력 객체타입으로 DPoint, DLineString, DPolygon 등을 정의했다. 그림 1은 OpenGIS 공간모델에 유효시간을 지원하도록 확장한 2차원 객체의 시공간 타입클래스의 계층도이다.



그림 1 2차원 객체의 시공간 클래스 계층도

2차원 공간데이터를 시간으로 확장하기 위해 유효시간을 표현하는 ValidTime 클래스를 정의하였으며, 시간과 관련된 모든 기본 연산자들이 포함된다.

TGeometry는 Geometry 클래스와 ValidTime 클래스로부터 이중 상속받은 2차원 시공간 객체에 대한 추상클래스이며, 시간

1. 서론

최근 GIS의 영역에서 중요한 연구분야 중 하나로 시간지원 지리정보 시스템의 개발이 대두되고 있다. 시간지원 지리정보 시스템은 데이터베이스에서 공간객체의 속성으로 시간차원이 포함된 GIS이다. 기존의 GIS에 시간속성을 추가하는 것은 공간객체의 영역이나 위치가 시간에 따라 변화하는 과정 또는 특정 시간에서 공간 객체들의 상태를 캡처하는 등의 공간적 분석이 가능하도록 해준다. 이러한 시간 및 공간에 대한 연산자는 위상 연산자, 기하 연산자, 기타 연산자로 분류하며, 이 중에서 가장 중요한 것은 공간 객체들간의 위상 관계를 처리하는 위상관계 연산자이다. 지도상의 객체들 간의 관계를 효율적으로 알아내는 것은 시스템의 성능을 향상시킬 뿐만 아니라, GIS 응용 프로그램을 개발하는데 필수적인 요소이다.

이 논문에서는, 컴포넌트 기반의 OLE/DB 인터페이스를 통하여 2차원 시공간 객체를 제공하기 위한 시간과 공간을 통합시킨 시공간 위상관계 연산자들을 설계한다. OGC의 OLE/DB를 위한 OpenGIS Simple Feature 명세서는 COM 기반의 OLE/DB 컴포넌트를 이용하여 공통의 표준 인터페이스를 제공한다[OGC99]. OGC에서 정의한 Geometry 클래스는 공간 기하객체를 나타내기 위한 최상위 추상 클래스이며 OLE/COM을 기반으로 하는 인터페이스를 제공한다. Point, LineString과 Polygon 객체들은 이러한 Geometry의 인터페이스를 상속받는다. TGeometry는 공간 객체에 유효시간 속성을 부여하기 위해 Geometry와 ValidTime이라는 유효시간 추상 클래스를 이중 상속받는다[이현아01]. TGeometry 클래스는 시공간 객체에 대한 최상위 추상 클래스이며 시간과 공간에 대한 통합 연산자들을 제공한다.

논문의 구성은 2장에서 시공간 객체 모델, 2차원 공간 객체에 대한 위상관계 연산자, 시간 위상관계 연산에 대해 다루고, 3장에서는 이들을 통합한 시공간 위상관계 연산들을 설계하고, 4장에서는 실제 적용 예를 설명하고 마지막 5장에서 결론을 맺는다.

[†] 이 논문은 한국전자통신연구원의 "4D 시공간 데이터 제공자 컴포넌트 개발" 연구지원으로 수행되었음.

과 공간에 대한 통합 개념을 제공한다. 이 논문에서는 TGeometry가 갖는 시공간 위상관계를 위한 tContains, tCrosses, tDisjoint, tEquals, tIntersects, tOverlaps, tTouches, 그리고 tWithin 연산을 설계한다.

2.2. 공간 연산자

공간 객체들이 이웃한 관계로서 정의되는 연속성, 폐쇄, 내부, 경계의 개념을 사용한다. 피쳐들은 지형객체들의 집합이고 또한 지형객체들은 점들로 이루어져 있기 때문에[Munk75], 피쳐와 그들간의 관계를 정의하기 위한 일반적인 틀을 제공하기 위하여 이러한 집단은 점들의 집합에 기반 한다. 우리는 보통 GIS에서 사용되는 피쳐들간의 관계성을 다루기 위하여, 객체들간의 위상 관계에 대한 연구는 2차원(\mathbb{R}^2)이라고 가정한다. 다음은 각 객체의 내부, 경계, 그리고 외부의 영역의 교차를 이용하여 연산들을 정의하는 방법으로 이 교차방법을 이용한다. 다음은 각각의 교차 방법을 설명한다. 또한 뒤에서 차원으로 확장된 9IM 방법을 이용하여 시공간에 관련된 8개의 연산들을 설계한다.

2.3. 차원으로 확장된 9IM (DE-9IM)

DEM 방법과 4IM 방법을 합한 방법으로써, 각각의 교차 집합들을 차원으로 확장하고 또한 내부, 경계 뿐만 아니라 외부도 고려한 방법이다[Clem95, OGC99]. 다음은 차원에 대한 교차행렬을 보여주고 있다.

$$M = \begin{pmatrix} \dim(\partial\lambda_1 \cap \partial\lambda_2) & \dim(\partial\lambda_1 \cap \lambda_2^o) & \dim(\partial\lambda_1 \cap \lambda_2^b) \\ \dim(\lambda_1^o \cap \partial\lambda_2) & \dim(\lambda_1^o \cap \lambda_2^o) & \dim(\lambda_1^o \cap \lambda_2^b) \\ \dim(\lambda_1^b \cap \partial\lambda_2) & \dim(\lambda_1^b \cap \lambda_2^o) & \dim(\lambda_1^b \cap \lambda_2^b) \end{pmatrix}$$

9개의 교차 집합이 가질 수 있는 값은 $\{-, 0, 1, 2\}$ 이기 때문에, DE-9IM 방법은 $4^9 = 262144$ 개의 가능한 경우의 수가 존재한다. 다음은 그림 2를 가지고 DE-9IM 방법을 이용한 예를 보인다.

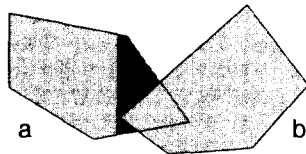


그림 2 교차방법을 위한 예

	내부	경계	외부
내부	2	1	2
경계	1	0	1
외부	2	1	2

각각의 값들은 객체의 내부(또는 경계나 외부)와 내부의 교차된 영역에 대한 차원이며, 그림 2에서 두 객체 a와 b의 교차된 영역은 2차원의 영역객체이기 때문에 2로 나타낸다. 각각의 교차 값들의 조합으로 두객체의 관계를 나타낼 수 있다. 그래서 위와 같은 패턴(212101212)을 갖으면 overlap이라고 정의할 수 있다.

$$\begin{aligned} p = T &\Rightarrow \dim(x) \in \{0, 1, 2\}, & i.e. x \neq \emptyset \\ p = F &\Rightarrow \dim(x) = -1, & i.e. x = \emptyset \\ p = * &\Rightarrow \dim(x) \in \{-1, 0, 1, 2\}, & i.e. Don't Care \\ p = 0 &\Rightarrow \dim(x) = 0 \\ p = 1 &\Rightarrow \dim(x) = 1 \\ p = 2 &\Rightarrow \dim(x) = 2 \end{aligned}$$

패턴행렬은 9개의 패턴 값으로 이루어진다. 가능한 패턴 값 p는 $\{T, F, *, 0, 1, 2\}$ 이고, 그들의 의미는 다음과 같다. 패턴행렬은 행 우선으로 9개 문자의 리스트나 배열로 표현되어

질 수 있다. 예로, 다음과 같은 코드는 두 개의 영역간의 overlap을 검사하는데 사용되어질 수 있다.

```
char * overlapMatrix = "T*T***T**";
Geometry* a, b;
Boolean b = a->Relate(b, overlapMatrix);
```

2.4. 시간 연산자

시간 위상관계는 공간에서의 위상관계와 비슷한 개념을 포함하는데, Allen은 before, after, starts, started_by, finishes, finished_by, during, contains, meets, met_by, equals, overlaps, overlapped_by들을 포함하는 13가지의 시간비교 연산자를 정의하였고[Alle83], Sarda는 이력지원 데이터베이스를 위한 시간 인스턴스 비교연산자, 시간간격 비교연산자, 시간간격 연산자, 시간집계 연산자를 정의하였고[Tans93], Snodgrass는 TSQL2에서 각 시간 데이터형(Datetime, Interval, Period)에 대해 산술 연산자, 비교 연산자, 집계함수 등을 정의하였다[Snod95].

3. 시공간 위상 관계 연산자

OLE/DB 컴포넌트를 위한 2차원 공간 데이터모델이 2차원 시공간 데이터모델로 확장되고, 기존의 2차원의 공간 데이터들을 다루기 위한 연산들이 모두 2차원의 시공간 데이터모델을 다루기 위하여 확장되어야한다. 이 논문의 목적은 2차원 시공간 데이터를 위한 연산들로 확장하는 것이 목적이다.

기존의 시공간 모델에 대한 연산자의 연구는 SQL 질의 문에 대한 조건(predicate)으로서 overlap, contain, equals를 제안했다[이종연99]. 이 논문은 이러한 아이디어를 배경으로 해서 OGC에서 제공하는 연산들을 설계한다.

[OGC99]에서는 OLE/DB 컴포넌트를 위한 contains, crosses, disjoint, equals, intersects, overlaps, touches, within과 같은 8개의 2차원 위상관계 연산자를 제공한다. 2차원 시공간 모델에 대한 시공간 위상관계 연산자를 설계하기 위해서, 8개의 공간 관계연산에 대한 시간관계를 먼저 정의한다. 다음은 시간 관계에 대한 8개의 위상관계의 정의이다.

- a. Contains(b) = (a.from ≤ b.from) and (a.to > b.to)
- a. Disjoint(b) = (a.from > b.to) or (a.to ≤ b.from)
- a. Equals(b) = (a.from = b.from) and (a.to = b.to)
- a. Intersects(b) = ((a.from ≥ b.from) and (a.from < b.to)) or ((a.to ≥ b.from) and (a.to < b.to))
- a. overlaps(b) = ((a.from > b.from) and (a.from ≤ b.to)) or ((a.to > b.from) and (a.to ≤ b.to))
- a. Touches(b) = (a.from = b.to) or (a.to = b.from)
- a. Within(b) = (a.from > b.from) and (a.to ≤ b.to)
- a. Crosses(b) ⇔ a. Intersects(b) (단, from ≤ to)

그러나, “두 개의 시간차원은 cross 될 수 없다”는 시간의 법칙 때문에, 시간에 대한 cross 관계는 정의할 수 없다. 이 논문에서는 시간에 대한 cross관계를 시간에 대한 intersects 관계와 같다고 정의한다. 다음 그림 3은 이러한 시간관계에 대한 예를 보인다. 두 개의 시간인터벌(기준 시간과 대상시간)이 있을 때, 두 개의 시간에 대한 관계는 그림 3과 같다. 단, 각각의 시간인터벌의 시작 시점(from)은 끝 시점(to)보다 앞선 시간이다. 이제, 2차원 시공간 모델에 OpenGIS에서 제안한 8가지 연산자들을 시공간 모델에 적용한다. 위에서 정의한 공간에 대한 8개의 연산과 시간에 대한 8개의 연산을 동시에 적용한다.

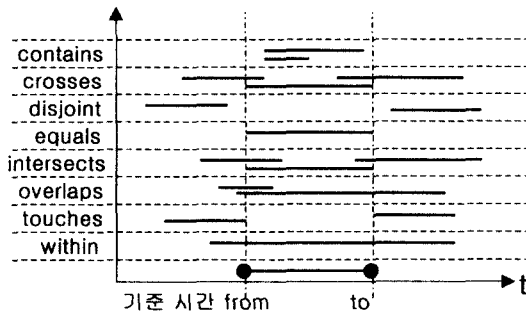


그림 3 시간에 대한 위상관계 연산의 예
다음은 이러한 시공간 연산자의 설계이다.

tOverlaps

주어진 두 개의 시공간 객체 a와 b에 대하여 교차되는가를 검사하는 연산자이다. 공간에 대한 overlap 연산은 영역/영역, 선/선, 그리고 점/점에 대한 연산만이 가능하다. 시공간에 대한 표현은 다음과 같다. 기타 다른 연산들 또한 다음과 같은 방법으로 정의된다.

선과 선에 대한 연산의 경우,

$$a.tOverlaps(b)$$

$$= (\dim(I(a) \cap I(b)) = 1) \wedge (I(a) \cap E(b) \neq \emptyset) \wedge (E(a) \cap I(b) \neq \emptyset) \wedge (((a.from > b.from) \wedge (a.from \leq b.to)) \vee ((a.to > b.from) \wedge (a.to \leq b.to)))$$

$$= a.Relate(b, '1* T*** T**') \wedge (((a.from > b.from) \wedge (a.from \leq b.to)) \vee ((a.to > b.from) \wedge (a.to \leq b.to)))$$

나머지 경우,

$$a.tOverlaps(b)$$

$$= (I(a) \cap I(b) \neq \emptyset) \wedge (I(a) \cap E(b) \neq \emptyset) \wedge (E(a) \cap I(b) \neq \emptyset) \wedge (((a.from > b.from) \wedge (a.from \leq b.to)) \vee ((a.to > b.from) \wedge (a.to \leq b.to)))$$

$$= a.Relate(b, 'T* T*** T**') \wedge (((a.from > b.from) \wedge (a.from \leq b.to)) \vee ((a.to > b.from) \wedge (a.to \leq b.to)))$$

위와 같이 나머지 tEquals, tContains, 그리고 tIntersects도 정의할 수 있으며, tContains는 tWithin과 정 반대의 개념이고, tIntersects는 tDisjoint와 정 반대의 개념이다.

4. 적용 및 평가

앞장에서 정의한 시공간에 대한 위상관계 연산자를 이용하여 실제 예에 적용한다. 그림 4는 실제 적용한 예를 보여준다. 그림은 x, y의 공간 차원과 t의 시간 차원을 다음과 같다고 가정한다. 각각 a와 b 객체는 TGeometry를 상속받은 TPolygon 객체이다. 이들은 물론, 앞에서 정의한 연산들을 갖는다. t1 시점에서 a1과 b1 객체가 각각 시간에 따라서 서로 다른 위치를 갖는다. 이들이 t2 시점에서 서로 교차하게 되고 t3 시점에서는 각각 다음과 같은 위치에 놓이게 된다. 만약, a 객체가 양떼들의 무리이고 b 객체는 폭풍 지역이라고 가정한다면, 다음과 같은 질의가 가능하다. "양떼들이 비에 젖는가?". 결과는 "t2 시점에 비에 젖는다"이다. 이것은 위에서 정의한 tIntersects 연산을 이용하여 해결한다. 위와 같이 시공간 데이터에 대하여 시간과 공간을 동시에 적용한 위상관

계 연산을 적용하였다. 또한 실제 적용에서는 다음과 같은 예들이 존재할 수 있다.

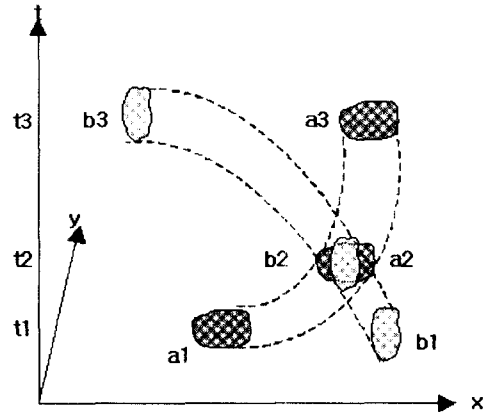


그림 4 시공간 객체의 위상관계 연산의 예

"태풍이 오전 5시에서 6시 사이에 어느 지역을 지나갔는가?", "A라는 그린벨트 지역에 태풍이 지나간 것들의 이력을 보여라.", "1990년도에 개신동에서 산불이 발생한 이력을 보여라", "소방차가 산불지역에 도착한 시간은 ?" 등등 많은 예들이 있을 수 있다. 이는 실제 응용에서 적용할 수 있다.

5. 결론

OGC는 2차원 공간데이터를 제공하기 위하여 OLE/DB 컴포넌트를 위한 표준인터페이스를 제공한다. OLE/DB는 마이크로소프트의 COM 기반 기술을 이용한 확장된 컴포넌트로서 표준 인터페이스를 갖는다. OGC에서는 이러한 OLE/DB 컴포넌트의 표준인터페이스를 이용하여 2차원 데이터를 제공하기 위한 방법을 제시하였다.

2차원 시공간 데이터모델에 대한 시공간 연산자를 정의하기 위하여, 공간에 대한 방법으로서 차원으로 확장된 9IM 방법을 이용하였는데, 이것은 공간 객체를 내부, 경계, 그리고 외부로 구분하고, 이것들간의 관계를 차원으로 나타내는 방법이다. 시간에 대한 방법으로서 OpenGIS에서 제안한 8개의 연산들을 시간에 적용했으며, 이들 두 가지 방법을 통합하여 시공간 객체에 대한 tContains, tWithin, tOverlaps, tIntersects, tTouches, tEquals, tDisjoint, tCrosses 연산들 설계하였다. 이러한 시공간 위상관계 연산자들은 OLE/DB의 ICommandWithParameter 인터페이스를 통해서 사용자의 질의의 조건으로 반영된다.

참고문헌

[Alle83] James F. Allen: Maintaining Knowledge about Temporal Intervals. CACM 26(11): 832-843, 1983.
 [Munk75] J. Munkres. Topology, A first Course. Prentice Hall, Englewood Cliffs, NJ, 1975.
 [류근호00] 류근호, 4D 시공간 데이터 제공자 컴포넌트 개발에 관한 연구, 한국전자통신연구소 최종보고서, 2000.
 [이현아01] 이현아, 김상호, 류근호, 2차원 지리 객체를 위한 시공간 객체 모델 설계, 정보처리학회 논문지, 2001(예정).
 [Clem95] E. Clementini and P. Di Felice. A Comparison of Methods for Representing Topological Relationships. Information Sciences 3: 149-178, 1995.
 [OGC99] Open GIS Consortium, Inc., OpenGIS Simple Feature Specification For OLE/COM Revision 1.1, OpenGIS Project Document 99-050, 1999.