

불확실한 시공간 객체에 관한 위상 관계 알고리즘

지 정 희[†] · 김 대 중^{††} · 류 근 호^{†††}

요 약

지금까지 명확하게 정의된 경계를 갖는 공간 및 시공간 객체 모델 개발에 관한 많은 연구가 수행되어 왔다. 그러나, 이들 모델은 지리 분석과 이미지 해석에 관한 많은 응용에서 식별되는 불확실한 경계를 갖는 공간 및 시공간 객체에 직접적으로 적용될 수 없다. 따라서, 이 논문에서는 불확실한 공간 및 시공간 객체에 적용할 수 있는 불확실한 시공간 데이터 모델을 제안하고, 이 모델을 기반으로 불확실한 시공간 객체간의 위상 관계에 관한 연산자를 정의하고, 연산 알고리즘을 설계하였다. 제안된 모델은 기존 모델과의 호환성을 위해 개방형 GIS 명세서를 기반으로 하는 시공간 데이터 모델을 확장하여 설계하였다. 불확실한 시공간 객체는 시간에 따라 위치와 모양이 불연속적으로 변하는 객체와 시간에 따라 위치와 모양이 연속적으로 변하는 객체로 정의하였으며, 확장된 9-IM을 사용하여 이들 객체간의 위상 관계를 정의하였다. 제안된 모델은 천연자원 관리시스템, 날씨 정보 관리 시스템, 지리 정보 관리 정보 시스템 등에 효율적으로 적용될 수 있을 것으로 기대된다.

Algorithm for Topological Relationship On an Indeterminate Spatiotemporal Object

Jeong Hee Chi[†] · Dae Jung Kim^{††} · Keun Ho Ryu^{†††}

ABSTRACT

So far, significant achievements have been studied on the development of models for spatial and spatiotemporal objects with precisely defined boundaries. However, these models can not be directly applied to spatial and spatiotemporal objects with indeterminate boundaries which are found in many applications for geographic analysis and image understanding. Therefore, in this paper we propose the spatiotemporal data model which is applicable for spatial and spatiotemporal objects with uncertainty. Based on this model, we defined topological relationships among the indeterminate spatiotemporal objects and designed the algorithm for the operations. For compatibility with existing spatial models, the proposed model has been designed by extending the spatiotemporal object model which is based on the open GIS specification. We defined indeterminate spatial objects, such as the objects whose position and the shape change discretely over time, and the objects whose shape changes continuously as well as the position. We defined topological relationships among these objects using the extended 9-IM. The proposed model can be efficiently applied to the management systems of natural resource data, weather information, geographic information, and so on.

키워드 : 시공간 데이터베이스(Spatiotemporal Database), 불확실성(Uncertainty), 불확실한 경계(Indeterminate boundary), Broad boundary, 위상관계(Topological Relationship)

1. 서 론

실세계의 복잡, 다양한 객체를 대상으로 하는 지리 및 공간정보 시스템들은 점차 대량의 데이터를 효율적으로 저장 관리하기 위해 분산환경에서 데이터베이스와 연계하여 개발되고 있다. 또한, 많은 응용분야에서는 시간에 따라 변하는 공간객체 즉, 시간과 공간을 통합한 시공간 객체를 다룰 수 있는 시공간 데이터베이스를 요구하고 있다. 최근 몇 년 동안 시공간 데이터베이스에 대한 관심이 고조되면서 효율

적으로 시공간 데이터를 모델링 하기 위한 많은 연구들이 수행되어 왔다[11, 21]. 이러한 연구들의 대부분은 기존의 관계형 데이터베이스를 확장하여 시공간 데이터를 모델링 하거나 객체지향 데이터베이스를 기반으로 시공간 데이터를 모델링하고 있다[19]. 이러한 연구들의 대상이 되는 공간객체는 길, 가옥, 필지 등과 같은 명확한 경계를 갖는 객체들이며, [10]에서는 이러한 객체들을 명확한 경계를 가진 공간객체라고 정의하고 있다. 현재 상용화된 많은 GIS 도구들은 이러한 공간객체 모델을 기반으로 하고 있다.

그러나, 명확한 경계를 갖는 공간객체만으로는 다양한 실세계 현상을 표현하기 어렵다. 예를 들면, 기존의 모델링 방식을 사용하여 태풍을 표현할 경우, 태풍은 단순한 원이나 다각형으로 표현되므로, 태풍 경계에 인접한 지역은 태

* 이 연구는 대학 IT 연구센터 육성, 지원사업과 한국과학재단 RRC(청주대 정보통신 연구센터)의 연구비지원으로 수행되었음.

† 준 회 원 : 충북대학교 대학원 전자계산학과

†† 준 회 원 : 국토연구원 GIS 연구 센터 근무

††† 종신회원 : 충북대학교 전기전자 및 컴퓨터공학부 교수

논문접수 : 2003년 2월 6일, 심사완료 : 2003년 7월 31일

풍의 영향권에서 벗어났다고 정의된다. 이러한 단순한 표현으로 인한 부정확한 정보의 생성 문제를 해결하기 위해서, 경계에 대해 버퍼링 기법이나 MBR(Minimum Bounding Rectangle)기법 등을 적용하는 연구들[6, 8, 10]이 수행되었다. 이들은 불확실한 경계를 갖는 공간객체를 불확실한(vague, indeterminate) 공간객체 또는 넓은 경계(broad boundary) 갖는 공간객체로 정의하고 있다. 이러한 연구를 기반으로 토지 속성의 변화, 인구 밀도, 토질 등과 같은 자연, 사회, 문화 현상, 생태 등을 보다 효율적으로 표현할 수 있게 되었다. 그러나, 기존 연구들은 불확실한 공간객체를 잘 표현하기 위한 방법만을 고려하고 있을 뿐, 불확실한 공간객체들의 시간에 따른 변화를 관리하기 위한 방법들은 고려하지 않고 있다.

따라서, 이 논문에서는 불확실한 공간객체들의 시간에 따른 변화를 표현하기 위한 불확실한 시공간 데이터 모델을 제안하고, 제안된 모델을 기반으로 불확실한 경계를 갖는 시공간 객체(이하 불확실한 시공간 객체)에 대한 위상관계 알고리즘을 설계한다. 이를 수행하기 위해 먼저, [21]의 단순한 경계를 갖는 시공간 데이터 모델을 수용하여, 시간에 따라 불연속적으로 변하는 불확실한 이력객체와 연속적으로 변하는 불확실한 이동객체를 정의한다. 그리고, 불확실한 시공간 객체들간의 분석기능을 제공하기 위해 시공간 위상관계 연산자를 제안하고, 그 알고리즘을 설계한다.

제안된 시공간 위상관계 연산자는 다양한 위상관계를 표현하기 위해 기존의 단순한 경계에 적용하였던 9-IM(Intersection Method)[6]을 불확실한 경계를 처리할 수 있도록 확장된 9-IM을 사용하여 정의한다. 이러한 교차방법을 이용하여 공간 위상관계와 시간 관계 비교 연산을 수행할 수 있는 연산 알고리즘을 설계한다. 실세계의 불확실한 공간객체는 대부분 영역으로 표현되는 객체이므로[5], 제안된 시공간 위상 연산은 불확실한 시공간 영역객체를 대상으로 한다. 제안된 연산은 시간에 따라 불연속적으로 변하는 이력 객체의 위상 관계 뿐만아니라, 시간에 따라 연속적으로 변하는 이동객체의 특정 시점에 관한 위상 관계 분석을 지원할 수 있다.

이 논문의 구성은 다음과 같다. 우선, 2장에서는 기존의 시공간 모델과 불확실한 경계를 가진 공간객체에 대한 관련 연구를 시작으로 3장에서는 기존의 시공간 데이터 모델의 공간 모델을 불확실한 경계를 가지는 공간 모델로 확장하여 시공간 데이터 모델을 제안하고, 4장에서는 이 논문에서 제안한 모델을 기반으로 불확실한 시공간 위상관계 연산 알고리즘을 설계한다. 또한, 5장에서는 제안한 불확실한 경계를 가진 시공간 모델과 연산자를 기반으로 프로토타입을 구현하여 그 타당성을 검증하고, 다른 모델과의 비교 평가를 한다. 마지막으로 6장에서는 이 연구의 결론 및 향후 연구를 기술한다.

2. 관련 연구

이 장에서는 불확실한 시공간 데이터 모델을 설계하기 위해, 기존에 연구된 시공간 데이터 모델과 불확실한 공간객체의 표현법에 대하여 기술한다.

2.1 시공간 데이터 모델

현재까지 제안된 시공간 데이터 모델들은 여러 가지 관점에서 연구되어 왔다. 예를 들어, 모델링 방법이 관계형, 객체지향 혹은 객체관계형 인지에 따라 서로 다르게 설계되었다. 또한, 객체의 속성이 정적인지 동적인지에 따라 다르고, 지원되는 시간 형태가 유효시간, 거래시간, 두 시간을 모두 지원하는 이원시간인지에 따라 다르게 설계되어 왔다. 먼저, 객체지향 접근 방법을 최초로 도입하여 시공간 데이터베이스를 설계한 것은 [17]이다. 객체지향 데이터베이스는 상속, 은폐, 다형성 등을 지원하므로 복잡한 구조를 갖는 데이터에 대한 관리가 용이한 반면에 관계형 데이터베이스는 관리 비용이나 연산 효율성 등에서 단점을 가지고 있다. 이 두 가지 기법의 장점을 이용하여 [17]은 공간 모델로부터 시공간 모델로 확장하였다. 시공간 객체는 공간과 이원시간을 포함하는 ST-Object의 개념을 도입하였다. 여기에서, 공간 객체는 점, 일직선, 그리고 삼각형의 Simplex들로 정의하고 있다. 이 Simplex에 대해 이원시간 타임스탬핑을 한 객체를 ST-Simplex라고 하고, 시공간 객체가 이것의 유한 집합인 ST-Complex에 의해 표현될 수 있음을 보였다. 또한, ST-Complex 상에서 합집합, 교집합, 차집합, 동등성, 부분집합, 공간 투영, 그리고 시간 투영 등의 대수들을 제안하였다. 그러나, 기본적으로 제한된 연산 집합을 갖는 ST-Complex라는 단일 타입만을 정의하므로 표현력이 제한되고 표준 모델과의 호환성을 보장하지 않는 문제점이 있다.

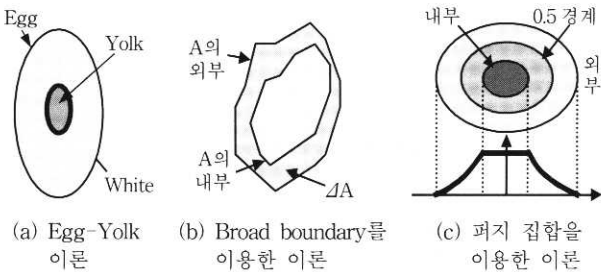
또한, [1, 2]는 관계형 데이터베이스를 기반으로 모델링하였으며, 유효시간과 트랜잭션 시간을 지원한다. 일반적으로, 시공간 데이터베이스는 이력에 따른 대용량의 데이터 그리고, 관계형 데이터베이스에서 제공하지 못하는 다양한 연산과 데이터 타입을 요구한다. 이를 위해 이 모델에서는 시공간 절의어 및 집계함수의 처리를 제안하였다. 그러나, 이 모델은 시간에 따라 변화하는 공간데이터의 모든 버전을 관리하므로 저장 공간의 낭비 및 시공간 표현이 부족한 단점을 갖는다.

그리고, 최근에는 시간에 따라 변하는 공간객체, 즉 이동객체의 개념을 이용하여 시공간 데이터베이스를 모델화[21, 11]하려는 연구가 진행되고 있다. [21]은 OpenGIS의 기하 객체 모델을 기반으로 이력객체와 이동객체를 모두 지원할 수 있도록 확장한 모델을 제안하였다. [11]에서는 연속적으

로 움직이고 있는 객체, 즉 시간에 따라 위치 또는 위치 및 모양 모두가 연속적으로 변하는 객체의 표현을 위하여 슬라이스 표현을 제안하였다. 즉, 한 객체의 시간에 따른 변화를 슬라이스라는 조각들로 재구성하며, 이 슬라이스의 값 변화를 함수로써 표현한다. 각 슬라이스들은 UINT 타입에 의해 표현되고 이 슬라이스들의 통합은 매핑 타입에 의해 이루어진다. 이 방법에 의해 시간에 따라 위치가 변하는 이동 라인 타입과 위치뿐만 아니라 모양 또는 영역이 변하는 이동 폴리곤 타입을 제시하였다. 이 논문에서는 기존 모델과의 호환성을 위해 [21]에서 제안한 시공간 데이터 모델을 기반으로 한다.

2.2 불확실한 공간 객체

기존의 많은 공간 데이터베이스 시스템 등에서 경계에 대한 일반적인 해석방법은 단순한 경계 즉, 단순한 선으로 가정하였다. 하지만, 시스템의 구축을 위한 데이터의 수집 단계에서부터 불확실성이 발생하므로, 경계에 대한 해석이 달라진다. 또한, 제한된 해상도나 다양한 변형 단계, 즉 서로 다른 자료의 변환이나 통합 등을 거치면서 데이터의 손실 등이 발생된다. [8, 15, 16]는 불확실한 공간객체를 크게 불확실성(uncertainty)과 모호성(fuzziness) 측면으로 구분하였다. 먼저, 불확실성은 객체의 정확한 측정이 불가능하거나 그 경계의 정확한 위치와 모양이 확실하지 않은 경우이며, 모호성은 객체가 임의의 영역을 가지고는 있으나, 그 경계가 명확하지 않은 경우를 나타낸다. 이러한 불확실한 공간 객체를 표현하기 위해, 공간의 축소와 단순한 경계를 가진 공간 기하객체의 표현(점, 선, 영역) 등을 이용하여 추상화하는 연구들이 수행되었다[5, 8, 9, 13]. (그림 1)(a)에서 보여주는 바와 같이 기존의 불확실한 경계에 대한 접근 방법 중 [9, 13]는 Egg-Yolk 이론을 제시하였다. 이 이론의 Yolk는 내부 요소를 나타내고, Egg의 흰 부분은 Egg와 Yolk간의 차집합으로 정의하고 있다. 또한, [5]는 (그림 1)(b)와 같이 Egg-Yolk 이론과 유사한 접근 방법으로 Egg의 흰 부분과 상응하는 불확실한 영역을 이용하여 불확실한 경계를 가진 공간객체를 표현하였다.



(그림 1) 불확실한 공간 객체에 대한 접근 방법들

그리고, [18]에서는 (그림 1)(c)에서와 같이 퍼지 집합 이

론을 기반으로 불확실한 경계를 가진 영역을 좀더 세부적으로 표현하고 있다. 그들은 임의의 불확실한 경계를 가진 영역에 포함된 각각의 점들을 소속정도에 따라 0과 1사이의 소속도 값으로 할당하고, 주어진 소속도 값들의 집합을 이용하여 영역의 내부 경계와 외부 경계를 정의하였다.

그러나, 여러 이론들이 불확실한 공간객체를 정의하고는 있지만, 그 정의가 한정되어 있고 구현의 어려움이 있다. 이런 단점을 극복하기 위해서 [8]에서는 정확한 모델(exact model)을 기반으로 불확실한 공간객체를 표현하였다. 이 방식은 기존의 단순한 공간객체 모델을 확장하여 불확실한 공간 객체에 대해 명확한 경계를 갖도록 표현함으로써 모호성 과 불확실성을 지원하고 있다. 이 논문에서는 [8]의 불확실한 공간객체 표현을 수용하였다.

3. 불확실한 시공간 데이터 모델

데이터베이스의 시간 형태는 유효시간, 거래시간, 사용자 정의시간등을 사용할 수 있다. 특히 유효시간과 거래시간은 현실 세계에서 발생하는 객체의 이력을 정확하게 기록할 수 있다. 여기서, 유효시간은 현실세계에 존재하는 객체에 대한 논리적인 시간을 나타내고, 거래시간은 데이터베이스에 기록된 시간을 나타낸다. 이 논문에서는 데이터베이스에 기록된 시간보다는 현실세계에 객체가 존재하였던 시간을 관리하기 위해 객체의 유효시간을 고려하였다.

제안하는 불확실한 시공간 데이터 모델은 기존 모델과의 상호 운용성을 증진시키기 위해 개방형 GIS 공간 모델에 유효시간을 지원하도록 확장한 시공간 모델을 기반으로, 불확실한 경계를 갖는 공간 및 시공간 객체를 지원할 수 있도록 설계하였다. 불확실한 공간객체를 나타내기 위해 BB Polygon을 정의하였으며, 시간에 따라 불연속적으로 변하는 불확실한 시공간 이력객체를 나타내기 위해 DBBPolygon을 정의하였다. 또한 시간에 따라 연속적으로 변하는 불확실한 시공간 이동객체를 나타내기 위해 MBBBPolygon을 정의하였다.

(그림 2)는 불확실한 경계를 지원하도록 확장한 불확실한 시공간 기하객체의 클래스 계층도를 나타낸다.

3.1 불확실한 공간객체

BBGeometry는 하위 클래스에 불확실한 경계에 대한 개념을 제공하기 위해 정의된 추상 클래스이다. 하위 클래스로서 BBPolygon이 존재하며, 기본 메소드는 <표 1>과 같으며, 불확실한 공간 객체간의 위상관계 연산은 <표 2>와 같다. 이에 대한 설명은 4장의 불확실한 시공간 객체간의 위상 관계 연산 기술서 언급한다. <표 2>의 α , β 는 불확실한 공간객체를 나타내며, $\Delta\alpha$, $\Delta\beta$ 는 불확실한 경계 영역을 나타낸다.

현하며, 유효시간의 시작점과 끝점이 동일한 DPolygon 객체로 구성된다.

[정의 3] $DBBPolygon \Leftrightarrow \{DP1, DP2\} = \{(P1, VTs, VTe), (P2, VTs, VTe)\}$

단, DP1 과 DP2는 DPolygon 객체이고, P1 과 P2는 Polygon 객체이다. VTs, VTe는 유효시간의 시작점과 끝점을 나타낸다. $P1.VTs = P2.VTs$ 이고 $P1.VTe = P2.VTe$ 이다.

MBBPolygon은 불확실한 공간 영역객체에 대한 시공간 객체로서 시간에 따른 불확실한 영역의 연속적인 변화를 표현하며, DBBPolygon을 상속받은 이동객체이다. 이력객체와 이동객체의 내부 자료구조가 비슷하며, 동일한 정보를 저장하고 있음을 의미하지만, 해석에는 차이가 있다. 이력객체는 이전 버전의 공간속성이 다음 버전까지 그대로 유지되는 반면, 이동 객체는 이전 버전부터 연속되는 다음 버전까지 공간속성이 계속 변하고 있기 때문에, 이러한 내부적 관계를 해석하기 위한 메소드가 필요하다. 이 논문에서는 이동 영역객체의 위치 및 영역을 추정하기 위해 [21]의 위치 및 영역 추정함수 $P(t)$ 를 BBPolygon의 내부와 외부 경계를 추정할 수 있도록 $P_I(t), P_E(t)$ 로 제정의함으로써 불확실한 이동 영역 객체의 위치와 영역 추정함수 $P_B(t)$ 를 정의하였다. $P_I(t), P_E(t)$ 함수의 반환값은 Polygon이며, 위치 및 영역 추정 함수 $P_B(t)$ 의 반환값은 DBBPolygon이다. 각각의 함수는 다음과 같이 나타낼 수 있다.

$$P_B(t) = (P_I(t), P_E(t)), t_k \leq t \leq t_{k+1}, \text{ 단 } P_I(t) \subseteq P_E(t) \quad (1)$$

$$P_I(t) = L_k(t), t_k \leq t = t_{k+1} \quad (2)$$

$$P_E(t) = L_k(t), t_k \leq t \leq t_{k+1} \quad (3)$$

여기서, $L_k(t)$ 는 [21]의 Polygon의 위치 및 영역을 추정하기 위한 함수이며, 부분구간 선형보간법을 사용하여 위치를 추정하고 있다. $P_I(t)$ 와 $P_E(t)$ 함수는 $L_k(t)$ 를 사용하여 Polygon 객체의 위치 및 영역을 추정한다.

4. 불확실한 경계에 대한 시공간 위상관계 연산 알고리즘

이 장에서는 불확실한 시공간 객체들 사이의 시공간 위상 관계를 분석하기 위한 연산자를 제안한다. 제안된 연산자는 공간 위상 관계를 불확실한 공간객체에 대해 지원하도록 확장한 9-IM 과 Allen의 시간 관계 연산자를 기반으로 한다.

4.1 불확실한 시공간 객체의 공간 위상관계 연산

일반적으로, 명확한 경계를 갖는 두 객체간의 공간 위상 관계를 표현하기 위한 방법으로 다양한 위상관계를 구별할

수 있는 포인트셋 위상(point-set topology)에 기반한 9-IM 을 많이 사용한다[4, 22].

그러나, 기존의 단순한 경계를 가진 모델에 적용한 9-IM 을 경계가 영역인 불확실한 경계에 적용할 수 없으므로, 단순한 경계를 불확실한 경계로 확장한 9-IM을 사용한다[7]. (그림 3)은 불확실한 경계에 대해 확장된 9-IM을 보여주고 있다. 여기서, $\Delta A, \Delta B$ 는 불확실한 시공간 객체의 경계 영역을 나타내며, A, B는 내부영역, A^o, B^o 는 외부를 나타낸다.

$$M = \begin{pmatrix} A^o \cap B^o \neq \phi & A^o \cap \Delta B \neq \phi & A^o \cap B^- \neq \phi \\ \Delta A \cap B^o \neq \phi & \Delta A \cap \Delta B \neq \phi & \Delta A \cap B^- \neq \phi \\ A^- \cap B^o \neq \phi & A^- \cap \Delta B \neq \phi & A^- \cap B^- \neq \phi \end{pmatrix}$$

(그림 3) 불확실한 경계에 대한 확장된 9-IM

위의 확장된 9-IM을 사용할 경우, 많은 위상 관계가 발생하므로, 이 논문에서는 [7]에서 분류한 연산중 대표적인 6개의 공간 연산을 이용하여 3.1절의 불확실한 공간객체간의 위상 관계를 정의하였다.

4.2 불확실한 시공간 객체의 시간 관계 연산

불확실한 시공간 객체간의 시간 관계를 분석하기 위한 연산은 <표 4>와 같다. 이 연산은 TGeometry 클래스에 정의되는 연산으로, 유효시간의 시작시간은 항상 유효시간의 끝 시간 보다 작거나 같아야 한다는 가정을 기반으로 한다.

<표 4> 시공간 객체에 대한 시간 관계 연산자들[3]

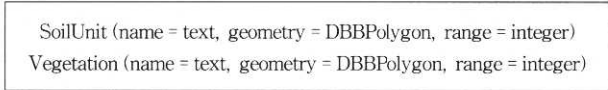
연 산 자	설 명
Before()	A.To < B.From일 경우 참
Equals()	(A.From = B.From) \wedge (A.To = B.To)일 경우 참
Meets()	A.To = B.From일 경우 참
Overlaps()	(A.To \leq B.To) \wedge (A.From \leq B.From)일 경우 참
During()	(A.From > B.From) \wedge (A.To < B.To)일 경우 참
Starts()	(A.From = B.From) \wedge (A.To < B.To)일 경우 참
Finishes()	(A.From > B.From) \wedge (A.To = B.To)일 경우 참

4.3 불확실한 경계에 대한 시공간 위상 관계 연산자

이 절에서는 불확실한 시공간 위상관계 연산을 정의하고 알고리즘을 설계한다. 불확실한 경계를 갖는 시공간 객체를 표현하기 위해 (그림 5)에서 보여주는 것과 같이 α 와 β 의 시공간 객체를 사용한다. α 는 유효시간을 갖는 두개의 Polygon α_1, α_2 로 구성되고, β 는 유효시간을 갖는 두개의 Polygon β_1, β_2 로 구성된다. $\partial \alpha_1, \partial \beta_1, \partial \alpha_2, \partial \beta_2$ 는 각각 $\alpha_1, \beta_1, \alpha_2, \beta_2$ 의 경계로 α, β 의 내부경계와 외부경계를 나타낸다. 또한, $\Delta \alpha, \Delta \beta$ 는 α, β 의 불확실한 경계를 나타낸다.

이 논문에서 제안한 시공간 연산자를 사용자 질의에 적용시키기 위해 사용되는 시나리오는 토질과 식생분포를 대상

으로 하였으며, 이용되는 추상 스키마는 (그림 4)와 같다.



(그림 4) 예제 질의를 위한 스키마 구조

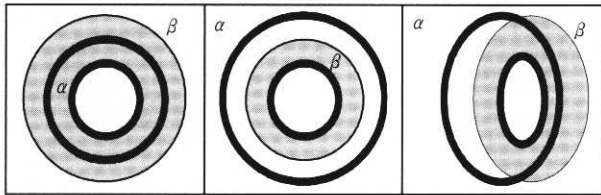
각각의 시공간 객체인 SoilUnit과 Vegetation은 DBBPolygon 개체이며, range는 하나의 속성을 갖는 범위값이다.

질의 예를 설명하기 위해 1980년 1월~12월까지의 토질 분포를 나타내는 데이터베이스에서 A 토질(SoilUnit.range = 3 : 찰흙)과 B 토질(SoilUnit.range = 4 : 이관암)이 존재하고, C 식생(Vegetation.range = 3 : 백합)과 D 식생(Vegetation.range = 4 : 장미)이 존재한다고 가정한다.

[정의 4] BVnearlyEqual (α, β)

$$((\alpha.from \leq \beta.from) \wedge (\alpha.to \leq \beta.to) \wedge (\alpha^\circ = \beta^\circ) \wedge (\Delta\alpha \cap \Delta\beta \neq \phi))$$

α 의 유효시간은 β 의 유효시간에 대해서 겹침(Overlap)의 관계가 성립하고, α 의 내부(α°)와 β 의 내부(β°)에 동등관계가 성립하면서 α 와 β 의 불확실한 경계 영역 사이에 교차가 발생하는 경우 참을 반환한다. (그림 5)는 BVnearlyEqual 연산중 공간 관계를 나타내고 있다.



(그림 5) BVnearlyEqual의 공간 관계

위의 연산자는 포함(Contain) 연산자 중 두 객체간의 내부 경계가 일치하는 좀더 세밀한 관계를 알고자 하는 경우 적용 가능하며, (그림 4)의 스키마 구조를 이용하여 (질의 1)과 같은 사용자 질의에 적용 가능하다.

(질의 1) “1980년 3월부터 10월 동안에 D 식생에 속하는 객체 중 A 토질과 거의 같은 영역에 분포되었던 객체를 검색하여라.”는 아래와 같은 질의 구문을 통해 나타낼 수 있다.

```

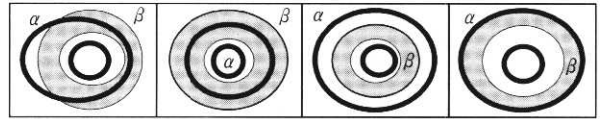
SELECT Vegetation.name
FROM SoilUnit, Vegetation
WHERE BVnearlyEqual (Vegetation, SoilUnit) AND
      SoilUnit.range = 3 AND Vegetation.range = 4 AND
      Vegetation.from ≥ 1980.03 AND Vegetation.to ≥ 1980. 10 ;
    
```

[정의 5] BVnearlyFill (α, β)

$$((\alpha.from \leq \beta.from) \wedge (\alpha.to \leq \beta.to) \wedge (\alpha^\circ \cap \beta^\circ \neq \alpha^\circ) \wedge (\Delta\alpha \cap \beta^\circ \neq \phi) \wedge (\Delta\alpha \cap \Delta\beta \neq \phi))$$

α 의 유효시간은 β 의 유효시간에 대해서 겹침관계가 성립하고, α 의 내부는 β 의 내부에 포함되면서, α 의 $\Delta\alpha$ 는 β 의 β° 와 $\Delta\beta$ 과 교차가 일어나는 경우 참을 반환한다. (그림 6)은 BVnearlyFill 연산중 공간 관계를 나타내고 있다.

위의 연산자는 포함 연산자중 두 객체간의 내부 경계에 대한 포함관계를 알고자 하는 경우 적용가능하며, (질의 2)과 같은 사용자 질의에 적용 가능하다.



(그림 6) BVnearlyFill의 공간 관계

(질의 2) “1980년 3월부터 10월 동안 D 식생에 속하는 객체 중 B 토질 영역 내에 거의 포함되도록 분포되었던 객체를 검색하여라.”는 아래와 같은 질의 구문을 통해 나타낼 수 있다.

```

SELECT Vegetation.name
FROM SoilUnit, Vegetation
WHERE BVnearlyFill (Vegetation, SoilUnit) AND SoilUnit.range = 4 AND Vegetation.range = 4 AND Vegetation.from ≥ 1980.03 AND Vegetation.to ≤ 1980.10 ;
    
```

[정의 6] BVstronglyOverlap (α, β)

$$((\alpha.from \leq \beta.from) \wedge (\alpha.to \leq \beta.to) \wedge (\alpha^\circ \cap \beta^\circ \neq \phi) \wedge (\alpha^\circ \cap \Delta\beta \neq \phi) \wedge (\Delta\alpha \cap \beta^\circ \neq \phi) \wedge (\Delta\alpha \cap \Delta\beta \neq \phi))$$

α 의 유효시간은 β 의 유효시간에 대해서 겹침관계가 성립하고, α, β 의 내부가 서로 겹치면서, α, β 의 경계가 서로 교차하는 경우에 참을 반환한다.

위의 연산자는 겹침 연산자중 두 객체간의 내부가 겹침관계에 대한 세부 관계를 알고자 하는 경우 적용가능하며, (질의 3)과 같은 사용자 질의에 적용 가능하다.

(질의 3) “1980년 2월에서부터 8월까지의 기간 중에 B 토질과 B 토질의 번이지대에 C 식생과 C 식생의 번이지대가 공존했었던 C 식생의 객체를 검색하여라.”는 아래와 같은 질의 구문을 통해 나타낼 수 있다.

```

SELECT Vegetation.name
FROM SoilUnit, Vegetation
WHERE BVstronglyOverlap (Vegetation, SoilUnit) AND
      SoilUnit.range = 4 AND Vegetation.range = 3 AND
      Vegetation.from ≥ 1980.02 AND Vegetation.to ≤ 1980.08 ;
    
```

[정의 7] BVboundaryOverlap (α, β)

$$((\alpha.from \leq \beta.from) \wedge (\alpha.to \leq \beta.to) \wedge (\Delta\alpha \cap \Delta\beta \neq \phi) \wedge (\alpha^\circ \cap \beta^\circ \neq \phi) \wedge (\alpha^\circ \cap \Delta\beta \neq \alpha^\circ) \wedge (\beta^\circ \cap \Delta\alpha \neq \beta^\circ))$$

α 의 유효시간은 β 의 유효시간에 대해서 겹침관계가 성립

하고, α 와 β 의 내부는 교차하지 않으며, α 의 내부가 β 의 경계에 포함되며, α 의 경계가 β 의 내부를 포함하는 경우에 참을 반환한다.

위의 연산자는 겹침 연산자중 두 객체간의 내부경계에 대한 비겹침관계를 알고자 하는 경우 적용가능하며, (질의 4)와 같은 사용자 질의에 적용 가능하다.

[질의 4] “1980년 2월에서부터 8월까지의 기간 중에 A 토질의 변이지대에 C 식생의 변이지대가 존재하였던 C 식생의 객체를 검색하여라.”는 아래와 같은 질의 구문을 통해 나타낼 수 있다.

```
SELECT Vegetation.name
FROM SoilUnit, Vegetation
WHERE BVboundaryOverlap (Vegetation, SoilUnit) AND
SoilUnit.range = 3 AND Vegetation.range = 3 AND
Vegetation.from ≥ 1980.02 AND Vegetation.to ≤ 1980.08 ;
```

[정의 8] BVcoveredByBoundary (α, β)
 $((\alpha.from \leq \beta.from) \wedge (\alpha.to \leq \beta.to) \wedge (\alpha^\circ \cap \beta^\circ \neq \phi)$
 $\wedge (\alpha^\circ \cap \Delta\beta \neq \alpha^\circ) \wedge (\alpha^- \cap \beta^\circ \neq \phi)$

α 의 유효시간은 β 의 유효시간에 대해서 겹침관계가 성립하고, α 의 내부가 $\Delta\beta$ 에 포함되면 참을 반환한다.

위의 연산자는 교차 연산자중 두 객체간의 불확실한 경계에 대한 교차관계를 알고자 하는 경우 적용가능하며, (질의 5)과 같은 사용자 질의에 적용 가능하다.

[질의 5] “1980년 3월~10월 동안, A 토질의 변이지대에 존재했던 C 식생의 객체를 검색하여라.”는 아래와 같은 질의 구문을 통해 나타낼 수 있다.

```
SELECT SoilUnit.name
FROM SoilUnit, Vegetation
WHERE BVcoveredByBoundary (Vegetation, SoilUnit) AND
SoilUnit.range = 3 AND Vegetation.range = 3 AND
Vegetation.from ≥ 1980.03 AND Vegetation.to ≤ 1980.10 ;
```

[정의 9] BVcoverWithBoundary (α, β)
 $((\alpha.from \leq \beta.from) \wedge (\alpha.to \leq \beta.to) \wedge (\alpha^\circ \cap \beta^\circ \neq \phi)$
 $\wedge (\alpha^\circ \cap \beta^- \neq \phi) \wedge (\Delta\alpha \cap \beta^\circ \neq \beta^\circ))$

α 의 유효시간은 β 의 유효시간에 대해서 겹침관계가 성립하고, α 의 $\Delta\alpha$ 가 β 의 내부를 포함하면 참을 반환한다.

위의 연산자는 교차 연산자중 두 객체간의 불확실한 경계에 대한 교차관계를 알고자 하는 경우 적용가능하며, (질의 6)과 같은 사용자 질의에 적용 가능하다.

[질의 6] “1980년 1월~12월 동안, A 토질에 C 식생의 변이지대가 존재하는 C 식생의 객체를 검색하여라.”는 아래와 같은 질의 구문을 통해 나타낼 수 있다.

```
SELECT Vegetation.name
FROM SoilUnit, Vegetation
WHERE BVcoverWithBoundary (Vegetation, SoilUnit) AND SoilUnit.range = 3 AND Vegetation.range = 3 AND Vegetation.from ≥ 1980.01 AND Vegetation.to ≤ 1980.12 ;
```

4.2. 불확실한 시공간 위상 관계 알고리즘

이 절에서는 앞에서 정의된 6개의 불확실한 시공간 위상 관계 연산자 중 BVnearlyFill 등 3개의 연산자에 대한 알고리즘을 기술한다. 나머지 4개의 위상 관계 연산자는 이 절에서 기술되는 알고리즘과 비슷한 방식으로 기술될 수 있다.

```
BVnearlyFill (sourceObject, targetObject)
input : sourceObject, targetObject는 불확실한 경계를 가진 시공간 객체.
output : sourceObject와 targetObject가 BVnearlyFill 관계이면 참, 그렇지 않으면 거짓을 반환.
Boolean BVnearlyFill (sourceObject, targetObject)
{
  get a temporal element of sourceObject and targetObject
  : 시간 객체는 From과 To element를 가지며, Object.From = Object.To
  if (sourceObject.From = targetObject.From) and (sourceObject.To = targetObject.To)
  {
    get a spatial element of sourceObject and targetObject
    : tIBoundary, tBBoundary, tEBBoundary 메소드 이용.
    Generate a Intersection Matrix between sourceObject and target Object
    if ( Matrix [sourceObject.Interior] [targetObject.Interior] is “T” ) and
    ( Matrix [sourceObject.Interior] [targetObject.BroadBoundary] is “F” ) and
    ( Matrix [sourceObject.BroadBoundary] [targetObject.Interior] is “T” ) and
    ( Matrix [sourceObject.BroadBoundary] [targetObject.BroadBoundary] is “T” )
      return true ;
    else
      return false ;
  }
  else
    return false ;
}
```

(알고리즘 1) BVnearlyFill 연산

(알고리즘 1)은 BVnearlyFill에 대한 알고리즘이다. 입력받은 시공간 객체의 시간 관계에 대해 겹침관계가 성립하는지 계산한 다음, 공간 요소들간의 관계를 비교하여 행렬을 구성한다. 이 알고리즘은 기존의 Equal 관계보다 경계에 대한 세밀한 관계를 표현할 수 있으며, 입력되는 두 객체 α, β 간의 위상관계를 나타내는 Matrix[α .BroadBoundary][β .Exterior]과 Matrix[α .Exterior][β .BroadBoundary]가 임의의 값을 갖는다.

(알고리즘 2)는 BVstronglyOverlap에 대한 알고리즘이다. 입력받은 시공간 객체의 시간 관계에 대해 겹침관계가 성립하는지 계산한 다음, 공간 요소들간의 관계를 비교하여 행렬을 구성한다. Overlap 관계에서는 두 객체간의 모든 요소가

교차를 한다. 그러나, BVstronglyOverlap과 BVboundary Overlap 연산자는 Overlap 연산자와 비슷하지만, 두 객체간의 위상관계를 나타내는 Matrix[α .BroadBoundary][β .Exterior]와 Matrix[α .Exterior][β .BroadBoundary]가 임의의 값을 가지므로, 좀더 다양한 위상관계의 표현이 가능하다.

```

BVstronglyOverlap (sourceObject, targetObject)
input : sourceObject, targetObject는 불확실한 경계를 가진 시공간 객체.
output : sourceObject와 targetObject가 BVstronglyOverlap 관계이면 참, 그렇지 않으면 거짓을 반환.

Boolean BVstronglyOverlap (sourceObject, targetObject)
{
    get a temporal element of sourceObject and targetObject
    : 시간 객체는 From과 To element를 가지며, Object.From = Object.To
    if (sourceObject.From ≤ targetObject.From) and (sourceObject.To ≤ targetObject.To)
    {
        get a spatial element of sourceObject and targetObject
        : tBBoundary, tBBoundary, tEBBoundary 메소드 이용.
        Generate a Intersection Matrix between sourceObject and targetObject
        if ( Matrix [sourceObject.Interior][targetObject.Interior] is "T" )
            and
            ( Matrix [sourceObject.Interior][targetObject.BroadBoundary] is "T" ) and
            ( Matrix [sourceObject.BroadBoundary][targetObject.Interior] is "T" ) and
            ( Matrix [sourceObject.BroadBoundary][targetObject.BroadBoundary] is "T" )
                return true ;
            else
                return false ;
        }
    else
        return false ;
}
    
```

(알고리즘 2) BvstronglyOverlap 연산

```

BVcoverWithBoundary (sourceObject, targetObject)
input : sourceObject, targetObject는 불확실한 경계를 가진 시공간 객체.
output : sourceObject와 targetObject가 BVcoverWithBoundary 관계이면 참, 그렇지 않으면 거짓을 반환.

Boolean BVcoverWithBoundary (sourceObject, targetObject)
{
    get a temporal element of sourceObject and targetObject
    : 시간 객체는 From과 To element를 가지며, Object.From ≤ Object.To
    if (sourceObject.From ≤ targetObject.From) and (sourceObject.To ≤ targetObject.To)
    {
        get a spatial element of sourceObject and targetObject
        : tBBoundary, tBBoundary, tEBBoundary 메소드 이용.
        Generate a Intersection Matrix between sourceObject and targetObject
        if ( Matrix [sourceObject.Interior][targetObject.Interior] is "F" )
            and
            ( Matrix [sourceObject.Interior][targetObject.Exterior] is "T" )
            and
            ( Matrix [sourceObject.BroadBoundary][targetObject.Interior] is "T" ) and
    
```

```

( Matrix [sourceObject.BroadBoundary][targetObject.BroadBoundary] is "T" )
and ( Matrix [sourceObject.BroadBoundary][targetObject.Exterior] is "T" )
        return true ;
    else
        return false ;
}
else
    return false ;
}
    
```

(알고리즘 3) BVcoverWithBoundary 연산

BVcoveredByBoundary과 BVcoverWithBoundary 연산은 입력 받은 시공간 객체의 시간관계에 대해 겹침관계가 성립하는지 계산한 다음, 공간 요소들간의 관계를 비교하여 행렬을 구성한다. BVcoveredByBoundary과 BVcoverWith Boundary 연산은 서로 역관계에 있다.

5. 실험 및 평가

이 장에서는 제안된 불확실한 경계를 가진 시공간 데이터 모델 및 연산자의 적용 가능성을 보이기 위해 우리나라 여름, 가을에 자주 발생이 되어 수산업에 피해를 주는 적조 현상에 적용하여 실험하였다. 우선, 이 논문에서 불확실한 경계를 가진 시공간 객체에 대한 표현 기법으로 MBR 기법을 사용하여 불확실한 경계를 표현하였다.

5.1 구현 환경

이 논문에서 구현한 불확실한 경계를 가진 시공간 모델은 Window 2000 Server 환경에서 프로그램 개발 툴인 Visual C++ 6.0을 이용하여 구현하였으며, 데이터베이스는 SQL Server 2000을 사용하였다.

5.2 데이터 스키마

<표 5> Fish 테이블의 스키마

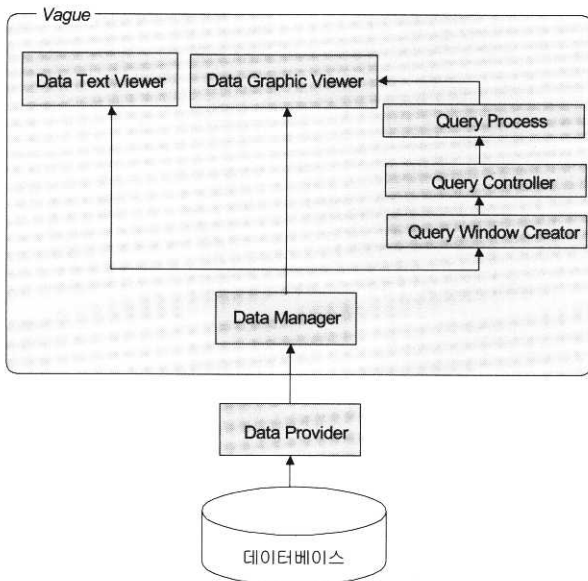
속성이름	데이터형	설 명
fid	int	id 값
fish	text	fish의 종류
innerX1	float	내부경계에 대한 MBR의 왼쪽 하단 좌표값
innerY1	float	
innerX2	float	내부경계에 대한 MBR의 오른쪽 상단 좌표값
innerY2	float	
outerX1	float	외부경계에 대한 MBR의 왼쪽 하단 좌표값
outerY1	float	
outerX2	float	외부경계에 대한 MBR의 오른쪽 상단 좌표값
outerY2	float	
From	Datetime	Valid time의 시작값
To	Datetime	Valid time의 종료값

데이터 셋은 국립 수산 과학원(NFRDI)에서 제공하는 2002년 8월 31일에 발생하여 2002년 9월 7일에 사라진 적조 데이터를 이용하여, 하루 단위의 시간 간격으로 데이터를 생성하였으며, 데이터 스키마는 아래와 같다. 여기에서 불확실한 경계는 내부 경계와 외부 경계로 표현하였다. <표 5>는 양식장에 대한 테이블의 정보를 나타내고, <표 6>은 적조 현상에 대한 테이블 정보를 나타내고 있다.

<표 6> Red_tide 테이블의 스키마

rid	int	id 값
type	text	적조의 종류
inner_density	float	내부의 밀도
outer_density	float	불확실한 경계의 밀도
innerX1	float	내부경계에 대한 MBR의 왼쪽 하단 좌표값
innerY1	float	
innerX2	float	내부경계에 대한 MBR의 오른쪽 상단 좌표값
innerY2	float	
outerX1	float	외부경계에 대한 MBR의 왼쪽 하단 좌표값
outerY1	float	
outerX2	float	외부경계에 대한 MBR의 오른쪽 상단 좌표값
outerY2	float	
From	Datetime	Valid time의 시작값
To	Datetime	Valid time의 종료값

5.3 구현 아키텍처



(그림 7) 구현 아키텍처

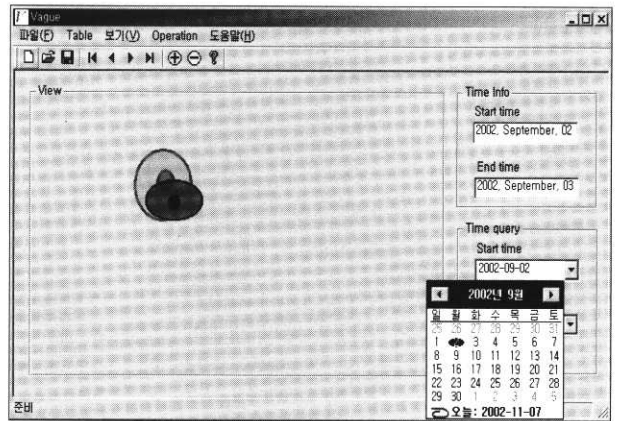
이 논문에서 구현한 프로토타입인 Vague는 (그림 7)과 같은 구현 아키텍처를 갖는다. 이 아키텍처는 크게 3부분으로 구성된다.

- Data Manager : DBMS인 SQL Server에 있는 데이터 셋은 이 모듈을 통하여 각각 데이터 셋에 접근한다.
- Data Viewer : Data manager에 의해 접근된 데이터를 스키마 정보를 볼 수 있게 테이블 형태로 보여주고, 또한 GUI를 통하여 동시에 두 개의 테이블 데이터를 보여준다.
- Query Process : 제안된 연산자 처리를 위해 윈도우가 생성되고, GUI에서 질의 내용을 입력 받아 질의 분석기와 질의 처리기에서 질의를 처리한다. 연산결과는 참(1)과 거짓(0)인 불린 값으로 반환된다.

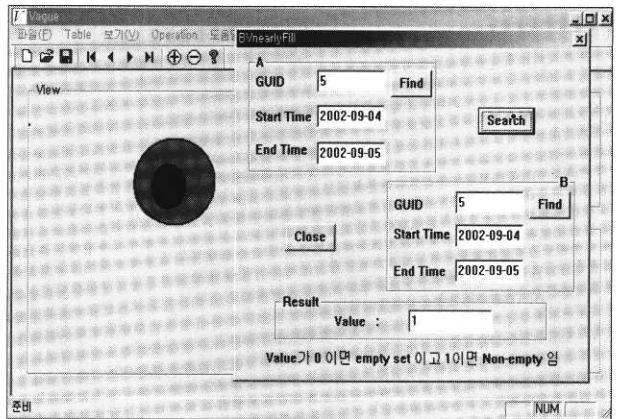
5.4 구현 및 수행

구현된 프로토 타입에서 수행한 사용자 전체 화면은(그림 8)(a)에서 보여주고 있다. 이 부분에서는 데이터의 내부 경계와 외부 경계에 해당하는 MBR 정보를 가져와 GUI를 통하여 화면에 보여주며, 화면의 왼쪽 하단 부분에서는 간단한 시간 질의를 가능하게 하였다.

(그림 8)(b)는 이 논문에서 제시한 불확실한 경계에 대한 6가지 연산자 중 BVnearlyFill 연산자를 구현한 화면이다.



(a) 전체 화면



(b) BVnearlyFill 연산자 구현 화면

(그림 8) 구현 결과 화면

아래와 같은 질의를 사용자로부터 GUI를 통해 입력받으면 두 개의 시공간 객체에 대한 시간, 공간 정보를 가져와서 질의처리기에서 질의 처리 과정을 거친 다음, 결과값으로 불린 값을 반환한다.

질의 : 2002년 9월 4일부터 2002년 9월 5일까지 밀집한 어종의 서식분포가 적조에 의해 거의 전역이 피해를 입었던 어종은 무엇인가?

위의 질의를 처리하기 위한 SQL의 WHERE 절 이하는 다음과 같이 나타낼 수 있다.

WHERE BVnearlyFill (Fish, Red_tide) AND Red_tide.From ≥ 2002.09.04 AND Red_tide.To ≤ 2002.09.05

이때, 처리 결과 값이 0이면 두 시공간 객체간에 주어진 교차관계에 대한 교집합이 없다는 것을 의미하고, 1이면 교차관계가 성립함을 보여준다. 이와 같은 수행 절차는 4장에서 제시한 BVnearlyFill 시공간 위상관계 연산 알고리즘에 따라 실행된다.

5.5 기존 연구의 비교 및 평가

이 절에서는 제안된 불확실한 시공간 데이터 모델의 타당성을 검사하기 위해 아래와 같은 기준을 선정해서 기존의 시공간 모델과 비교, 평가하며, 평가 기준은 다음과 같다.

- 기준 1 : 2차원 공간객체를 관리할 수 있는가?
- 기준 2 : 유효시간을 지원하는가?
- 기준 3 : 이력객체를 관리할 수 있는가?
- 기준 4 : 거래시간을 관리할 수 있는가?
- 기준 5 : 불확실한 개념을 지원하는가?
- 기준 6 : 위치와 영역이 변하는 이동객체를 관리할 수 있는가?
- 기준 7 : 이력객체의 시공간 연산을 지원하는가?
- 기준 8 : 위치가 변하는 이동객체의 시공간 연산을 지원하는가?
- 기준 9. 개방형 모델과 호환되는가?

<표 7>은 위의 기준에 따라 이 연구에서 제시한 결과와 다른 모델과의 비교 평가한 결과를 보여주고 있다.

<표 7>은 위와 같은 9가지의 기준에 대해 기존에 제안된 [8, 11, 17]의 시공간 모델과 이 논문에서 제안한 불확실한 경계를 가진 시공간 모델을 비교한다. 여기에서 주어진 기준을 만족할 때 ○, 기준에 대한 내용이 제한적으로 언급되어있는 경우는 △, 기준을 만족하지 못하였을 때는 ×로 표시한다. 이런 기준에 의해 [11, 17]은 9가지 중에 5가지 정도의 기준을 만족하였고, [8]는 3가지 기준을 만족하였다.

따라서, 불확실한 경계를 가진 시공간 모델은 8가지의 기준을 만족시킴으로써 다른 3가지 시공간 모델보다는 사실적인 표현 및 효율적인 시간 관리가 가능함을 알 수 있다.

<표 7> 기존의 시공간 모델과 불확실한 시공간 데이터 모델의 비교

기준 \ 모델	Worboys [17]	Forlizzi [11]	Clementini [8]	불확실한 경계를 가진 시공간 데이터 모델
기준 1	○	○	○	○
기준 2	○	○	×	○
기준 3	○	×	×	○
기준 4	○	△	×	×
기준 5	×	×	○	○
기준 6	×	○	×	○
기준 7	○	×	×	○
기준 8	×	△	×	○
기준 9	×	△	○	○

6. 결 론

기존의 공간 데이터베이스 시스템은 단순한 공간데이터 타입으로 실세계 객체를 표현하고 있다. 이 과정에서 공간객체의 단순화, 추상화는 실세계의 복잡, 다양한 객체를 표현하기에는 한계가 있고, 이미 데이터화 된 자료들간의 변환이나 통합 등에 의해 공간데이터의 불확실성이 발생하고 있다. 또한, 많은 분야에서 시간에 따른 객체를 관리하는 시공간 데이터베이스를 필요로 하고 있다. 따라서, 이 논문에서는 불확실한 경계를 갖는 시공간 객체, 즉 불확실한 시공간 객체에 관한 데이터 모델과 연산자를 제안하였다.

제안된 불확실한 시공간 모델은 기존 모델과의 상호운용을 위해서 개방형 GIS에서 제시한 2차원 기하객체 모델을 시간차원으로 확장한 시공간 데이터 모델을 기반으로, 불확실한 공간객체를 지원토록 확장하였으며, 또한 불확실한 시공간 객체를 지원토록 확장하여 설계되었다.

이 모델은 불확실한 시공간 이력객체와 이동객체를 지원하며, 불확실한 이동객체의 위치와 영역을 추정하기 위해 기존 부분구간 선형보간법을 사용하는 함수를 확장하여 정의하였다. 또한, 확장된 9-IM을 기반으로 불확실한 시공간 객체에 대한 시공간 위상 관계 연산자 및 알고리즘을 설계하였다.

제안된 불확실한 시공간 모델은 경계가 명확하지 않은 시공간 객체들간의 다양한 분석기능 특히, 위상 관계를 분석하여 제공해 줄 수 있으므로, 부정확한 정보를 제공

했던 기존 모델의 문제점을 보완할 수 있을 것이라고 기대된다.

향후 연구로는 불확실한 단순 시공간 객체 이외의 홀을 포함한 복잡한 불확실한 시공간 객체에 대한 모델 설계와 그에 따른 연산을 확장하는 연구와 불확실한 이동객체의 시간에 따른 위상관계 변화를 다루기 위한 연구가 진행될 것이다.

참 고 문 헌

- [1] Dong Ho Kim, Keun Ho Ryu and Hong Soo Kim, "A Spatiotemporal Database Model and Query Language," *Journal of Systems and Software*, Vol.55, No.2, pp.129-149, 2000.
- [2] Dong Ho Kim, Keun Ho Ryu and Chee Hang Park, "Design and Implementation of Spatiotemporal Database Query Processing System," *Journal of Systems and Software*, Vol.60, No.1, pp.37-49, 2002.
- [3] F. J. Allen, "Maintaining Knowledge About Temporal intervals," *Communications of the ACM* 26, pp.832-843, 1983.
- [4] E. Clementini, P. Di Felice and P. Van Oosterom, "A Small of Formal topological Relationships Suitable for end-User Intersection," In 3rd Int. Symp. On Advances in Spatial Databases, LNCS 692, pp.277-295, 1993.
- [5] E. Clementini and P. Di Felice, "Approximate topological relations," *International Journal of Approximate Reasoning*, Vol.16, pp.173-204, 1997.
- [6] E. Clementini and P. Di Felice, "spatial operator," *SIGMOD Rexord*, Vol.29, No.3, pp.31-38, 2000.
- [7] E. Clementini, P. Di Felice and K. Koperski, "Mining multiple-level spatial association rules for objects with a broad boundary," *Data & Knowledge Engineering*, Vol.34, No.3, pp.251-270, 2000.
- [8] E. Clementini and P. Di Felice, "A spatial model for complex objects with a broad boundary supporting queries on uncertain data," *DKE*, Vol.37, No.30, pp.285-305, 2001.
- [9] A. G. Cohn and N. M. Gotts, "The 'egg-yolk' representation of regions with indeterminate boundaries," In P. A. Burrough and A. U. Frank, editors, *Geographic Objects with Indeterminate Boundaries*, Taylor & Francis, pp.171-187, 1996.
- [10] Martin Erwig and Markus Schneider, "Vague region," 5th Int. Symp. On Advances in Spatial Databases (SSD '97), LNCS 1262, pp.298-320, 1997.
- [11] L. Forlizzi, B. H. Gting, E. Nardelli and M. Schneider, "A Data Model and Data Structures for Moving Object Databases," *SIGMOD Conference 2000*, pp.319-330, 2000.
- [12] D. J. Kim, J. H. Chi, S. H. Kim, S. J. Lee, K. H. Ryu and In-Hak Joo, "A Spatio-temporal model for simple objects with an indeterminate boundary," *ICIS2002, 2nd ACIS*, pp. 767-773, 2002.
- [13] F. Lehmann and A. G. Cohn, "The egg yolk reliability hierarchy : Semantic data integration using sorts with prototypes," In Proc. 3rd Int Conf. on Information and Knowledge Management, pp.272-279, Gaithersburg, 1995.
- [14] OpenGIS Consortium, "OpenGIS Simple Features Specification for SQL," 1998.
- [15] M. Schneider, "Uncertainty Management for Spatial Data in Databases : Fuzzy Spatial Data Types," 6th Int. Symp. on Advances in Spatial Databases (SSD), LNCS 1651, Springer Verlag, pp.330-351, 1999.
- [16] Roy, A. J. and J. G. Stell, "Spatial Relations between Indeterminate Regions," *Journal of Approximate Reasoning*, Smith, B. On Drawing Lines on a Map. *Spatial Information Theory - A Theoretical Basis for GIS*, COSIT95. A. U. Frank and W. Kuhn. Semmering, Austria, 2000.
- [17] M. F. Worboys, "A Unified Model for Spatial and Temporal Information, *The Computer Journal*, Vol.37, No.1, pp.26-34, 1994.
- [18] F. B. Zhan, "Approximate analysis of binary topological relations between geographic regions with indeterminate boundaries," *Soft Computing*, Vol.2, pp.28-34, 1998.
- [19] 김동호, 류근호, "관계형 시공간 데이터베이스 질의언어와 연산", *정보처리학회논문지*, 제5권 제10호, pp.2467-2478, 1998.
- [20] 김대중, 지정희, 류근호, "불확실한 경계를 가진 시공간 객체의 모델 설계", *정보처리학회 춘계학술발표논문집*, 제9권 제1호, pp.83-86, 2002.
- [21] 이현아, 남광우, 류근호, "2차원 지리 객체를 위한 시공간 객체 모델 설계", *정보처리학회논문지*, 제9권 제1호, 2002.
- [22] 지정희, 김대중, 류근호, "OGC 기반의 위상관계 연산 알고리즘, *한국정보과학회 데이터베이스연구회 특집논문집*", 제18권 제1호, pp.29-41, 2002.



지 정 희

e-mail : jhchi@dblab.chungbuk.ac.kr

1998년 충주대학교 전자계산학과

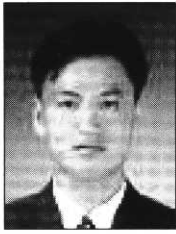
2001년 충주대학교 대학원 전자계산학

전공(이학석사)

2001년~현재 충북대학교 전자계산학과

박사과정수로

관심분야 : 시공간 데이터베이스, Temporal GIS, 시공간 질의 최적화, 시공간 색인기법, 이동객체 관리 기법



김 대 중

e-mail : djungkim@krihs.re.kr
1998년 공주대학교 지질환경학과
2003년 충북대학교 대학원 전자계산학
전공(이학석사)
2003년~현재 국토연구원 GIS 연구 센터
근무

관심분야 : 시공간 데이터베이스, 시공간 객체의 불확실성, GIS



류 근 호

e-mail : khryu@dblalab.chungbuk.ac.kr
1976년 숭실대학교 전자계산학과
1980년 연세대학교 공학대학원 전산전공
(공학석사)
1988년 연세대학교 대학원 전산전공(공학
박사)

1986년 육군군수지원사전산실(ROTC 장교), 한국전자통신연구소
(연구원), 한국방송통신대, 전산학과(조교수) 근무
1989~1991년 Univ. of Arizona 연구원(TempIS Project)
1986년~현재 충북대학교 전기전자 및 컴퓨터공학부 교수
관심분야 : 시간 데이터베이스, 시공간 데이터베이스, Temporal
GIS, 객체 및 지식베이스 시스템, 지식기반 정보검
색시스템, 데이터 마이닝, 데이터베이스 보안 및 Bio-
Informatics