

차량 궤적 추적을 위한 불확실성 처리기 구현

김진석* · 김동호** · 류근호***

요 약

인터넷 기술의 발달로 다양한 응용분야의 컴퓨팅 환경은 상당한 변화를 겪어 왔다. 특히 지난 3년여 동안 물류분야에서는 물류 회사간 인터넷 기반의 가상 비즈니스 활동 또는 서비스 아키텍처를 의미하는 e-로지스틱스라는 새로운 개념에 대한 활발한 연구가 진행되었다. 효과적인 e-로지스틱스 프레임워크 구축을 위해서 GIS, GPS 및 시공간 데이터베이스를 포함하는 이동체 기술 개발이 진행 중에 있다. 이동체 기술은 자동차, 비행기, 선박 등과 같이 시간에 따라 공간상의 위치를 변경하는 시공간 객체에 대한 효율적인 데이터 관리를 의미한다. 그러나 기존의 관련 시스템은 여러 가지 이유로 이동체에 대하여 오직 가장 최근에 검출된 위치정보만을 관리하기 때문에, 과거 및 미래의 불확실한 위치 추정 방법에 대한 구체적인 제시는 미약한 상태에 있다. 따라서 이 논문에서는 이동체에 대한 이력정보 관리와 이를 이용한 과거의 위치 추정이 가능한 시스템을 제안한다. 이를 위해 물류 운송차량 위치 추적을 위한 모델링과 데이터베이스 및 시스템 구조를 제시한다. 아울러 제안 시스템을 e-로지스틱스 분야의 주요 응용인 택배 차량에 대한 위치정보 불확실성 처리 시나리오에 적용하는 예를 가지고 설명한다.

Implementation of Uncertainty Processor for Tracking Vehicle Trajectory

Jin-Suk Kim* · Dong-Ho Kim** · Keun-Ho Ryu***

ABSTRACT

Along the advent of Internet technology, the computing environment has been considerably changed in many application domains. Especially, a lot of researches for e-Logistics have been done for the last 3 years. The e-Logistics means the virtual business activity and service architecture among the logistics companies based on the Internet technology. To construct effectively the e-Logistics framework, researches on the development of the Moving Object Technology(MOT) including GPS and GIS with spatiotemporal databases technique so far has been done. The Moving Object Technology stands for the efficient management for the spatiotemporal objects such as vehicles, airplanes, and vessels which change continuously their spatial location along with time flows. However, most systems manage just only the location information detected lately by many reasons so that the uncertainty processing for the past and future location of the moving objects is still very hard. In this paper, we propose the moving object uncertainty model and system design for e-Logistics applications. The MOMS architecture in e-Logistics is suggested and the detailed explain of sub-systems including the uncertainty processor of moving objects is described. We also explain the comprehensive examples of MOMS and uncertainty processing in Delivery Parcel Application that is one of major application of e-Logistics domain.

키워드 : 이동체(Moving Object), 불확실성 처리(Uncertainty Processing), e-로지스틱스(e-Logistics), 차량 추적(Vehicle Tracking)

1. 서 론

차량 위치추적 시스템(Vehicle Location Tracking System)은 다양한 유형의 차량 위치를 탐지하는 장치를 장착한 차량으로부터 다양한 방법으로 전송받은 위치 정보를 토대로 현재 차량의 위치 정보나 교통량의 흐름 등을 실시간으로 제공해 주는 시스템으로 정의할 수 있다.

이때 추적 대상이 되는 차량은 시간의 흐름에 따라 그 위치 정보가 연속적으로 변경되는 특성을 가짐에도 불구하고 기존의 차량 위치 추적을 위해 개발된 많은 시스템들은 이

동체의 위치 데이터에 대하여 매우 제한된 범위만을 저장하였다. 즉, 개별적인 차량의 가장 최근 데이터만을 대상으로 서비스를 제공하였다.

최근 들어 인터넷을 기반으로 하는 관련 기술과 비즈니스 영역이 확장 및 발전함에 따라 이전에 개발된 시스템에 기능 및 성능을 추가하고 보완하는 필요성이 제기되어 왔으며, 이를 해결하기 위한 연구들이 진행되어 왔다.

대표적으로 이동통신기술의 발전과 관련 서비스의 발전으로 이동 중인 차량에서 관심 대상체를 질의하면 대응하는 결과를 제공하는 위치 기반 서비스(Location-based Service : LBS)와 국가 교통 인프라 구축을 위해 차량의 위치, 교통 및 운전과 관련된 체계적인 종합 서비스를 의미하는 지능형 교통 시스템(Intelligent Transport System : ITS)을 들 수 있다.

* 정 회 원 : 한국전자통신연구원 우정기술연구센터 책임연구원
 ** 중 신 회 원 : 한국전자통신연구원 우정기술연구센터 선임연구원
 *** 중 신 회 원 : 충북대학교 전기전자컴퓨터공학부 교수
 논문접수 : 2003년 7월 1일, 심사완료 : 2004년 8월 27일

이들 서비스와 시스템에서는 차량의 위치를 검출하는 측위기술을 포함하는 요소기술들을 필요로 한다. 뿐만 아니라 이동체 위치 데이터에 대한 효율적인 저장 및 처리 방안을 포괄하는 이동체 데이터베이스(Moving Object Databases: MODB)에 관한 연구도 진행되어왔다. 이러한 이동체에 대한 연구는 구체적으로 데이터 모델링, 질의 표현, 인덱스 구조, 불확실성 처리 방법 등이 있으며, 차량 위치 추적과 비즈니스에 관련된 응용 시스템들의 개발에 관한 노력도 초기 진행 단계에 있다.

이동체 위치 추적을 위한 데이터는 갱신 간격이 매우 짧고 빈번하게 발생하는 매우 거대한 분량을 갖는 일종의 시공간 데이터(Spatiotemporal Data)이다. 이러한 이유로 지금까지 진행된 대부분의 연구에서는 관리의 효율과 성능의 관점에서 이동체의 위치 데이터에 정보를 제한적으로 관리하고 있다. 즉, 기존의 차량추적 관련 연구들은 방대한 데이터에 대한 효과적인 관리 방안을 갖지 못한 이유로 오직 가장 최근에 검출된 이동체 위치 데이터만을 관리하기 때문에, 데이터베이스에 저장되지 않는 과거 및 미래의 불확실한 위치 추정 방안은 미약한 상태에 있다.

따라서 이 논문에서는 이와 같은 문제점을 해결할 수 있도록 위치정보의 기반의 차량 궤적정보 불확실성 처리기를 제안한다. 세부적으로 관심 대상인 차량의 위치 데이터에 대한 불확실성 처리 모델을 제시하고 및 불확실성 처리 구현을 통하여 차량 궤적 관리의 효율을 개선하고자 한다. 보다 효과적인 설명을 위해 세부적으로 다음과 같이 논문을 구성하였다. 먼저 2장에서는 개발 시스템과 관련된 이전 연구들을 정리하며, 3장에서는 운송차량 위치 추적을 위한 이동체 모델링과 데이터베이스 및 시스템 구조를 제시한다. 4장에서는 본 논문에서 제안하는 불확실성 처리 모델과 시스템 구조를 설명한다. 또한 e-로지스틱스 분야의 주요 응용인 택배 차량에 대한 위치정보 불확실성 처리 시나리오에 적용한 예는 5장에서 제시하고, 6장에서는 본 연구의 요약과 향후 진행방향을 설명한다.

2. 관련 연구

이동체(Moving Object)는 시간에 따라 객체의 공간 정보가 연속적으로 변경되는 시공간 데이터로 이동점과 이동영역으로 구분된다. 이동점(Moving point)은 시간에 따라 객체의 위치가 변하는 것으로 이동 점 객체에는 사람, 동물, 자동차, 비행기, 배 등이 있다. 이동영역(Moving region)은 시간에 따라 객체의 위치뿐만 아니라 모양까지 변하는 것으로 한 국가의 행정 구역이나 폭풍의 영향권, 암세포의 상태 등을 예로 들 수 있다. 이와 같은 이동체를 저장 및 관리하는 이동체 데이터베이스는 위치 기반 서비스, 차량 위치 추적, 물류 관리 등을 위한 기본 소프트웨어 인프라로 인식되

고 있다[1-4].

이동체 관리를 위한 대표적인 응용 시스템 연구에는 DOMINO, CHOROCHRONOS, DEDALE, Battlefield Analysis가 있다. DOMINO[5-9]는 이동 객체의 수송에 대한 웹 기반 실시간 궤도 응용의 개발을 촉진하는 이동 객체 소프트웨어 도구로서, DBMS 기술을 활용하여 만든 실시간 위치 추적 시스템 프로토타입이다. 그러나 DOMINO 프로토타입은 이동 객체의 현재 위치, 속도, 방향정보를 이용하여 미래의 이동 위치를 예측하는 방법에 주로 초점을 맞추고 있다. 따라서 과거 시점을 포함하는 이동 객체의 완전한 이동 경로인 궤적(trajjectory)을 관리할 수 없는 단점을 가진다.

CHOROCHRONOS[10, 11]는 시공간 데이터베이스의 특수한 형태인 이동 객체에 관한 연구가 집중적으로 수행되었고, 이동 객체의 데이터 모델링 및 인덱싱에 관한 연구 결과를 가장 많이 발표하였다. 특히 이 연구는 GPS 기반의 수송 관리 시스템과 멀티미디어 시스템에 적용한 응용 시나리오를 제시하였다. 그러나 아직 이동 객체 데이터베이스를 활용한 응용 시스템의 모델 및 개발 사례는 제시되지 않고 있으며, 개발 중인 질의처리 시스템에서는 이동 객체의 불확실한 과거 및 미래의 위치 정보 추정에 관한 방법이 제시되지 않고 있다.

DEDALE[11]은 제약사항 데이터베이스 모델을 이용하여 시공간 데이터를 모델링하고 질의처리를 하기 위해 개발된 프로토타입으로, 기존의 시공간 데이터의 모델뿐만 아니라 이동 객체의 궤적 등과 같은 데이터 모델 및 질의 표현도 제공하였다. DEDALE은 고급 수준의 개발 및 최적화를 위한 질의어의 확장을 위해, 공간 질의에 대한 최적 계산 기법의 사용 대신 기하 데이터에 대해 선형 제약사항 추상화를 제공하였다. 그러나 DEDALE 프로토타입은 데이터베이스에 시간의 변화에 따른 이동 객체의 (x, y) 좌표 값이 직접 저장되지 않고, 특정 구간의 궤적을 표현하는 선형 제약사항(linear constraint)의 공식이 저장된다. 이로 인해 빈번하게 이동하는 객체의 실시간 위치 추적을 위한 응용 시스템 개발에는 부적합한 특징을 가진다.

Battlefield Analysis[12-14]는 모의 전장에서 이동하는 부대 및 탱크들의 움직임을 예측하여 이를 의사결정에 활용할 수 있도록 개발된 전장분석 프로토타입이다. 전장분석 프로토타입은 이동 객체 관리기와 추론 엔진을 접목시키고자 하는데 초점이 맞추어졌다. 특히, 시공간 이동체의 연산 결과를 추론 엔진에서 활용하는 새로운 이동체 추론 모델을 제시하였다.

지금까지의 관련 연구를 분석해보면 이동체의 과거 및 미래의 위치정보를 하나의 데이터베이스에서 동시에 관리하지 못하며, 동일한 데이터베이스에 저장된 이력 위치정보를 이용하여 이동 객체의 불확실한 과거의 위치 추정 방법을 제시하지 못하는 문제점을 가지고 있다.

3. 차량 위치정보 모형

이 장에서는 차량 위치 추적을 위한 이동체 관리 시스템에서 차량의 위치에 대한 데이터 표현과 연산자를 포함하는 데이터베이스 모델[15]을 정리한다.

3.1 데이터 구조

2장에서 언급한 바와 같이 이동체는 형태에 따라서 이동점(Moving point)와 이동 영역(Moving region)으로 세분화된다. 이것은 이동체에 대한 데이터 표현 복잡도를 의미하는데, 이동 차량의 경우는 이동 점에 해당하며 다음과 같이 데이터 형태를 정의한다.

[정의 1] (이동체) 시간의 변화에 따라 객체의 위치 값만 변화되는 이동체를 말한다. 이동체 MP 는 시간 속성, 공간 속성, 일반 속성을 가지며, $MP = \langle T_A, S_A, G_A \rangle$ 가 된다. □

[정의 2] (시간 속성) MP 의 시간 속성 $T_A = \langle vt_s, vt_e \rangle$ 로 구성되며 vt_s 는 시작 시간, vt_e 는 종료 시간을 나타낸다. 이 때 vt_s 와 vt_e 는 유효시간(Valid time)의 집합 S_{VT} 의 원소가 된다. 유효시간은 실세계에서 발생된 시간을 나타내며, $S_{VT} = \{t_0, t_1, t_2, \dots, t_k, \dots, t_{now}\}$ 이고, 각 원소들은 $t_0 < t_1 < t_2 < \dots < t_k < \dots, \dots < t_{now}$ 의 순서를 가진다. $t_k = t_{k-1} + 1, t_k = t_0 + k, k \geq 0$ 인 정수로 정의된다. t_{now} 는 현재 시간을 의미하는 시간 상수이다. □

유효시간의 도메인은 선형 시간(Linear time), 이산 시간(Discrete time), 절대 시간(Absolute time)이며, 하나의 이동체 데이터베이스는 동일한 유효시간의 주기(Granularity)를 가진다. 여기서 동일 유효시간 주기란 차량의 위치에 대한 유효한 시간정보를 표현할 때 발생 또는 저장하는 간격을 의미하는데, 일반적으로 분 단위를 가장 많이 사용하며 비교적 세밀한 데이터 구축과 적절한 데이터 분량을 가진다.

[정의 3] (공간 속성) MP 의 공간 속성 $S_A = \langle x, y \rangle, x, y \in R$, R 은 실수이다. □

[정의 4] (이동체 데이터베이스) 이동체 데이터베이스를 구성하는 이동체들의 집합은 $S_{MP} = \{MP_0, MP_1, \dots, MP_n\}$ 이다. S_{MP} 로 구성된 이동체 데이터베이스의 이력 집합은 $H_{MP} = \{H_{MP_0}, H_{MP_1}, \dots, H_{MP_n}\}$ 이다. S_{MP} 에 속하는 각각의 MP_i 에 대한 모든 이력 집합은 $H_{MP_i} = \{mp_{i_0}, mp_{i_1}, \dots, mp_{i_n}\}$ 이고, $mp_{i_k} = \langle T_A(mp_{i_k}), S_A(mp_{i_k}), G_A(mp_{i_k}) \rangle$ 이다. □

[정의 4]에서 mp_{i_k} 는 MP_i 의 k 번째 이력 정보를 의미하고, $T_A(mp_{i_k})$ 는 MP_i 의 k 번째 시간 속성, $S_A(mp_{i_k})$ 는 k 번째 공간 속성, $G_A(mp_{i_k})$ 는 k 번째 일반 속성을 의미한다.

3.2 연산자

이동 차량의 위치정보 관리를 위해 다음과 같은 이동체 연산자를 정의한다. 차량의 일반 속성에 대한 검색은 기존의 상용 데이터베이스 시스템이 제공하는 기능을 그대로 사용하므로 별도로 정의하지 않는다. 이동체의 위치정보 관련 연산자로 $m_distance, trajectory, length, position_at, m_nearest, m_farthest$ 를 <표 1>과 같이 정의한다.

<표 1> 차량 위치 연산자

종류	입력 값	출력 값	기능
$m_distance$	$G_A \times G_A \times T_A$	Real	특정 유효시간 동안 두 이동 객체간의 거리 계산
$trajectory$	$G_A \times T_A$	{ S_A }	이동 객체의 특정 유효시간 동안의 이동 경로 추출
$length$	{ S_A }	Real	이동 객체의 특정 유효시간 동안의 이동 거리 계산
$position_at$	$G_A \times T_A$	S_A	임의의 한 시점에 대한 이동 객체의 위치 검색
$m_nearest$	$G_A \times T_A$	S_A	특정 유효시간 동안 가장 가까이 위치하는 객체 검색
$m_farthest$	$G_A \times T_A$	S_A	특정 유효시간 동안 가장 멀리 존재하는 객체 검색

<표 1>은 기존의 이동체 연구[5-8]에서 제시된 연산자들을 토대로 기본 연산자만을 정의한 것이다. 입력 및 출력 값에서 G_A 는 차량 소유주, 운전자, 차종 등과 같은 이동 객체의 일반 속성을 의미한다. S_A 는 공간 속성을 나타내고, T_A 는 시간 속성인 유효시간을 의미한다. Real은 실수 값을 나타내며, { }로 묶인 속성들은 집합을 의미한다.

4. 차량 위치정보 불확실성 처리

4.1 불확실성 모형

4.1.1 생성 유형

차량 위치정보 불확실성(Uncertainty)이란 임의의 이동체에 대하여 원하는 시간이 주어질 경우 여러 가지 이유로 대응하는 공간 위치데이터를 데이터베이스로부터 검색하지 못하는 경우를 방지하며, 동시에 이를 위해 요구되는 이동체 위치데이터의 분량을 최소화하기 위한 기술을 의미한다.

이것은 입력 질의에 대한 결과로서 반환되는 이동체의 위치 정확도를 향상시키기 위해서는 보다 빈번하게 데이터를

저장하여야 하는데, 실제 응용에서 차량 대수가 증가할수록 그 규모가 급격히 방대해지는 특성으로 인해 불확실성 처리를 제대로 지원하지 못하고 역으로 데이터베이스에 저장된 임의의 시간을 기준으로 질의하는 형태로 운용되고 있다.

차량의 위치에 관련된 불확실성을 유발하는 원인에는 ① 차량 위치정보 수집 시간간격으로 인한 유형, ② 통신 오류 또는 DB오류로 인한 유형, ③ 도로 및 교통정보 오류로 인한 유형, ④ 응용 분야 정보 오류로 인한 유형 등 크게 네 가지로 분류할 수 있다.

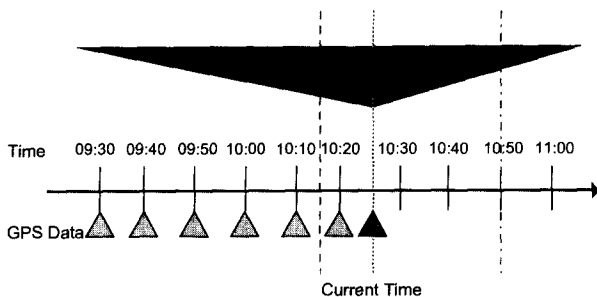
차량 위치정보 수집 시간간격으로 인한 유형은 이산적인 모델에 내재된 오류로서 이동체 관리 시스템(MOMS)의 성능과 비용에 막대한 영향을 미치는 중요한 요소이다. 이동체의 유형에 따라서 발생하는 위치정보의 오류 범위는 매우 다양하며, 이동체 위치정보 측위 장비의 정밀도와 통신망 속도 등과 같이 다양한 형태의 고려사항이 존재한다.

차량 위치정보를 유무선 통신망을 통하여 시스템의 데이터베이스에 저장하기까지는 수많은 통신 절차를 수행하게 되는데, 특히 무선 통신망의 경우는 주위 환경의 동적인 변화에 따른 영향으로 속도와 안정성에서 이동체의 위치정보에 대한 많은 불확실성을 야기 시킨다. 그리고 관련 시스템간의 표준화가 이루어지지 않는 환경에서는 개별적인 시스템의 데이터베이스에 저장된 차량 위치정보는 상호 호환성 관점에서 문제를 야기 시킬 수 있다.

도로 및 교통정보는 차량에 관련된 직접적인 정보로서 매우 밀접한 관련성을 가지며, 차량 위치정보를 위한 부가적인 데이터로서 활용될 수 있다. 일례로 일방통행여부, 제한 속도, 차선 등의 도로의 상세한 정보와 시간대별 교통량 통계 값을 들 수 있다. 하지만 때로는 이들 정보의 부정확성으로 인해 오히려 차량 위치정보의 불확실성을 증가시킬 수 있다.

4.1.2 질의 영역

차량 관리 기술에는 시간지원 데이터베이스(Temporal Databases) 개념을 반드시 포함하기 때문에 질의 형태를 크게 현재 시점을 기준으로 ① 과거의 위치정보 불확실성 질의(Retro-active Uncertainty Query)와 ② 가까운 미래의 위치정보 예측 불확실성 질의(Pro-active Uncertainty Query)로 구분할 수 있다.



(그림 1) 차량 위치정보 불확실성 분류

과거의 차량 위치정보 불확실성 질의는 데이터베이스 시스템에 저장된 차량 위치정보로부터 과거의 임의의 시간의 위치 정보를 추정하는 문제이다. 이 경우, 시공간 데이터베이스에는 해당 차량에 대하여 최소한 두 개 이상의 위치정보가 저장되어 있으며, 두 지점 사이의 임의의 위치 값을 추정하는 문제로 귀결한다. 또한 가까운 미래의 차량 위치정보 예측 불확실성 질의는 최대 허용 예측 시간범위 내의 미래에 대하여 최근에 획득된 차량 위치 정보를 토대로 추정하는 문제이다.

4.1.3 시공간 연산자 및 확장

차량 위치정보 불확실성 문제와 직접적으로 관련된 연산자에는 $AtTime(R_i, T_i)$ 가 있다. 연산자 $AtTime(R_i, T_i)$ 는 주어진 경로 R_i 에 대하여 입력된 시간값 T_i 에서의 위치값 P_i 를 반환한다. 식 (1)은 $AtTime(R_i, T_i)$ 연산에 대한 정형의미(Formal Semantics)를 관계 식으로 표현한 것이다.

$$AtTime(R_i, T_i) = \{P_k \mid \exists k ((P_k, T_k) \in R_i \wedge T_i \equiv T_k)\} \tag{1}$$

식 (1)에서와 같이 해당 연산은 시공간 데이터베이스에 저장된 위치 값 집합인 경로를 대상으로 검색을 하며, 만일 주어진 시간 T_i 와 일치하는 데이터가 없는 경우 $T_x < T_i < T_y$ 인 위치 값 P_x 와 P_y 를 대상으로 보간법(Interpolation)을 수행하여 추정된 값을 반환한다.

식 (2)는 불확실성 영역을 기반으로 보간법을 적용한 형태의 연산을 나타낸다. 보간법이란 함수 값을 알고 있는 주위 점들에 대한 가중평균을 이용하여 중간에 빠진 함수 값을 추정하는 방법을 의미하며, 대표적으로 선형 보간법(Linear Interpolation)은 두 점을 지나는 선분을 사용하여 값을 추정한다.

$$AtTime(R_i, T_i, Pr) = \begin{cases} P_k \mid \exists k ((P_k, T_k) \in R_i \wedge T_i \equiv T_k) & \text{if } (P_k) \neq null \\ P_k \mid P_k := UCR_{Pr}(R_i, T_i) & \text{otherwise} \end{cases} \tag{2}$$

4.1.4 불확실성 처리 모형

차량의 위치는 실시간으로 변경되기 때문에 모든 시간에 걸쳐 검출된 위치를 데이터베이스에 저장하는 것은 불가능하다. 따라서 차량의 위치는 특정 시간 구간을 갖고 이산적으로 데이터베이스에 저장된다. 하지만 이러한 데이터베이스는 저장되지 않은 시점의 차량 위치에 대한 질의에 대한 응답이 불가능하며 데이터베이스에 저장되지 않은 차량의 불확실한 이동 위치 정보에 대한 적절한 처리 방법이 필요하다.

차량의 불확실한 위치 추정은 다음과 같은 두 가지 위치 추정 모델을 사용한다. 첫 번째 모델은 선형 함수(Linear

function)를 이용한 위치 추정 모델이다. 선형 함수는 가장 단순한 1차 함수를 사용하므로 최소한의 연산으로 위치 추정 값을 구할 수 있는 장점을 갖는다. 두 번째 모델은 곡선을 표현하는 3차 스플라인 함수(Spline function)를 이용한 위치 추정 모델이다. 3차 스플라인 함수를 이용할 경우 일반적으로 선형 함수에 의한 위치 추정 결과보다 다소 정확한 값을 얻을 수 있으나 3차 함수를 구하기 위한 연산 처리 과정이 다소 복잡하므로 처리 속도가 지연되는 단점을 갖는다.

선형 함수는 차량의 이동 이력에 저장된 임의의 시간 간격의 위치 좌표를 이용하여 1차 함수를 구하고 이 함수를 이용하여 임의의 시점에 대한 위치 추정 값을 구한다. 선형 함수를 이용한 위치 추정 값의 근사는 차량의 이동 궤적을 직선 형태의 선형 궤적으로 모델링함을 의미한다.

선형 궤적 모델의 과거 위치 추정 연산은 식 (3)을 사용한다. 과거 위치 추정 연산은 특정 구간의 시작 시간, 시작 시간의 X, Y좌표 및 종료 시간, 종료 시간의 X, Y좌표를 사용하여 시간 구간의 변화 함수를 구하고 이 변화 함수에 질의 시점을 입력하여 질의 시점의 위치 좌표를 추정한다. 여기서 X, Y좌표는 각각 차량의 검출된 위치 값으로서 시스템에 인식되어지는 좌표체계상의 값을 의미한다.

$$x_{t_p} \leftarrow \frac{x_{t_{i+1}} - x_{t_i}}{t_{i+1} - t_i} (t_p - t_{i+1}) + x_{t_{i+1}}$$

$$y_{t_p} \leftarrow \frac{y_{t_{i+1}} - y_{t_i}}{t_{i+1} - t_i} (t_p - t_{i+1}) + y_{t_{i+1}}$$

(3)

예를들어, “차량의 ID가 1001인 객체의 유효 시간 23에서의 위치 값을 구하십시오” (단, 질의 구간의 시작 시점은 20, 이 때의 위치 좌표는 (125, 350)이고 종료 시점은 25, 이 때의 위치 좌표는 (145, 360)이다) 라는 질의를 보면 질의 시점은 23이 되고 질의 시점을 포함하는 최소 시간 구간은 [20, 25]가 된다. 과거 위치 추정 함수에 적용되는 위치 정보 값은 (20, 125, 350)과 (25, 145, 360)이 된다. 식 (3)에 의해 추정된 위치 값은 (137, 356)이 된다.

4.2 질의 유형 및 처리 방안

이동체 관리 시스템에서 불확실성 처리기(Uncertainty Processor)가 지원할 사용자 질의 유형과 대응하는 처리 방안은 다음과 같이 정리할 수 있다.

시점에 의한 차량의 위치 추적으로 “2002년 9월 4일 15시 15분 경 대전 80가 1234 차량의 위치를 확률 25%로 추정하라(기준시간은 2002년 9월 5일 10시 10분임)”라는 질의에 대하여 다음의 처리 방안을 들 수 있다.

- 입력 시간(2002년 9월 4일 15시 15분)을 포함하는 시간 간격에 대응하는 두 개의 차량 위치 데이터를 이동체 데이터베이스로부터 검색함.
- 추출된 두 지점으로부터 지정된 확률 값을 기반으로 불확실성 영역을 계산함.
- 계산된 불확실성 영역이 이전에 검색된 두 지점을 포함하는 지를 검사함.
- 포함되는 지점이 없는 경우 “해당 값 없음”을 반환함.
- 포함되는 지점이 하나인 경우 해당 값을 반환함.
- 포함되는 지점이 둘인 경우 입력 시간에 대응하는 지점으로부터 근접지점을 반환함.

<표 2> 불확실성 처리기의 데이터 정의

객체 명	객체 정의
데이터베이스	인덱스를 통하여 차량 데이터를 저장
입력 시점 구간 검색 모듈	데이터베이스에 저장된 시점 데이터 중 입력된 질의 시점에 가장 가까운 앞, 뒤 시간을 검색하는 함수
불확실한 위치 추정 모듈	차량 속도 검색 및 선형 보간 법을 사용하여 질의가 입력된 시점의 차량 위치 좌표를 추정
불확실한 영역 추출 모듈	차량의 위치 좌표와 입력된 불확실 비율을 가지고 불확실 영역 추출
불확실한 위치 확정 모듈	질의가 입력된 시점의 하루 전의 차량 데이터 중 불확실 영역 반경 내에 존재하는 차량을 검색하여 불확실한 위치로 확정

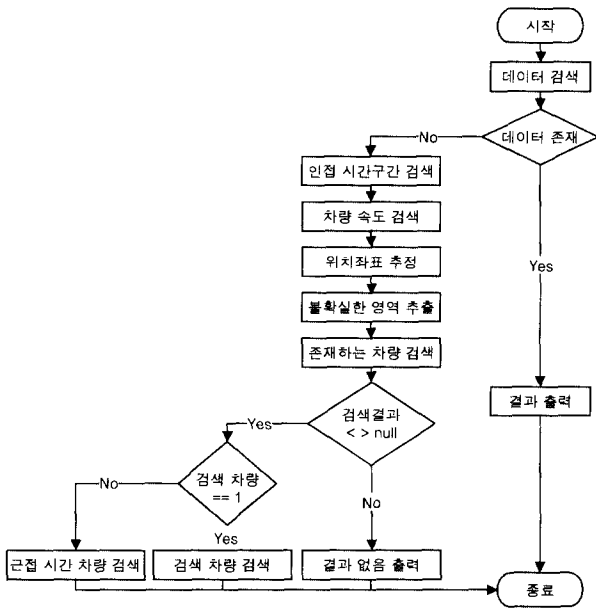
질의 시점에서의 차량 위치가 검색되면 그 위치를 반환한다. 검색되지 않는다면 질의 시점에 가장 가까운 시간 구간을 데이터베이스에서 검색한다. 또한 이 시간 구간의 차량 속도를 검색한다. 이를 검색하기 위한 질의 예는 다음과 같다.

질의 예) `select Vs, Ve, X_Vs, Y_Vs, X_Ve, Y_Ve Velocity from VehicleHistory where ID =id and Vs <= time and Ve > time`

그리고 입력된 질의 시점에서의 위치 좌표를 선형 보간법을 사용하여 추정한다. 위치 추정은 기존의 *PositionAtTime* 연산자에 존재하는 *Huncertain* 함수를 호출하여 실시한다.

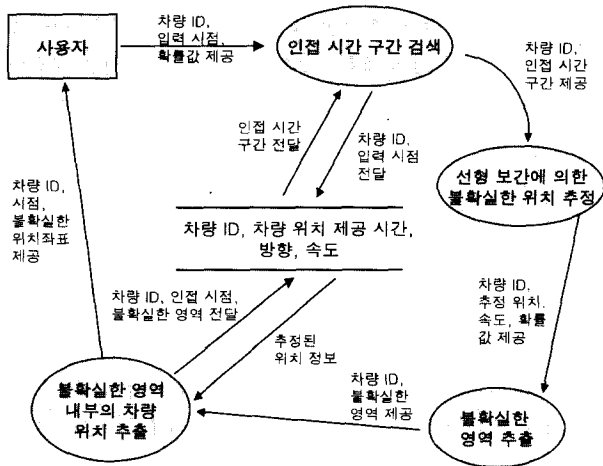
추정된 위치 좌표를 기반으로 전날 차량 위치 좌표 중에서 질의 시점과 가장 근접한 차량을 검색한다. 차량의 위치 검색은 사용자가 불확실한 시간 영역을 입력하고 입력된 시간 영역과 검색된 속도를 이용하여 구할 수 있다.

불확실한 영역을 구하기 위하여 불확실한 영역의 반경을 구해야 하고 그 반경은 ‘거리 = 속도 × 시간’이라는 공식에 의하여 구할 수 있다. 계산된 반경을 반지름으로 하는 원을 불확실성 영역이라고 하고 전날 차량 데이터 중 불확실한 영역에 존재하는 차량의 위치 데이터를 검색한다.



(그림 2) 이동체 위치정보 불확실성 처리 흐름도

전날 차량의 위치 데이터 중 불확실한 영역 내의 차량 위치 값을 검색한다. 이때, 이 영역 안에 위치 값이 없다면 “검색 차량 없음”이라고 출력한다. 반경 안에 하나의 위치 값이 있다면 그 위치 값을 반환한다. 반경 안에 두 개 이상의 위치 값이 검색되면 반경과의 거리를 비교하여 가까운 거리에 있는 불확실한 위치를 반환해 준다.



(그림 3) 이동체 위치정보 불확실성 처리 데이터 다이어그램

5. 시스템 설계

이동체 관리 엔진은 수집된 차량 위치 데이터에 대한 효율적인 저장 및 관리와 사용자로부터 입력되는 차량 위치 기반 질의에 대한 결과를 생성하는 핵심적인 기능을 제공한다. 즉, 이동체 관리 엔진은 기존의 전통적인 데이터베이스 관리 시스템의 기능에 공간 데이터 연산과 이력 데이터 연

산을 동시에 지원하는 특수한 형태의 시공간 데이터베이스 관리 시스템의 핵심 모듈이다.

5.1 시스템 구조

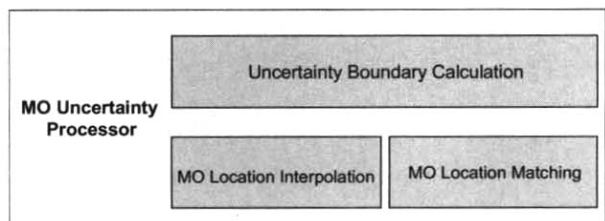
차량 위치정보 불확실성 처리 의미는 다음과 같이 정한다. 이동체 위치정보 불확실성의 기본 단위는 %로서 0~100 사이의 값으로 하고, 이들 값은 샘플링 되는 차량의 위치정보의 시간간격으로 매핑한다.

이동체 관리 시스템으로 입력되는 차량 위치정보의 샘플링 주기가 5분 간격으로 지정된 경우 25%는 75초(1분 15초)가 되며, 이동체의 구간 속도가 60Km/h였다면, 불확실성 영역은 그 지점으로부터 주행방향으로 1.25Km가 된다. 또한 40Km/h와 80Km/h의 경우에는 각각 0.833Km와 1.666Km이 된다.

차량의 속도 및 지정된 확률 값에 대응하는 검색 영역이 결정되면, 이를 기반으로 하는 데이터베이스로부터 위치 이력 데이터 값을 검색하는데, 두 개 이상의 데이터가 검색되는 경우에는 추출된 데이터 중에서 검색 영역 경계선에 가장 인접하게 존재하는 데이터를 우선적으로 선택한다. 사용자가 차량 위치정보 불확실성 확률 정도를 0으로 지정한 경우(기정의 값임)에는 가장 최근에 입력되어 저장되었거나 과거에 샘플링에 의해 저장된 값만을 대상으로 연산을 수행하며, 지정된 시간 값과 저장된 위치정보의 시간 값이 완전히 일치하지 않는 경우에는 “해당 값이 없음”으로 메시지를 출력한다.

불확실성 처리기는 (그림 4)에서 보여진 바와 같이 ① 불확실성 영역 검출(Uncertainty Boundary Calculation), ② 이동체 위치 보간(Moving Object Location Interpolation), ③ 이동체 위치 매칭(Moving Object Location Matching) 등의 모듈로 구성된다.

불확실성 영역 검출 모듈에서는 질의에서 제공하는 확률 값과 검색 대상인 차량의 정보로부터 속도 및 방향을 토대로 존재 가능한 불확실성 영역을 계산해 낸다. 이동체 위치 보간 모듈에서는 불확실성 영역 내에 정의되는 차량의 최소 두 위치에 대하여 이들 간의 연속적인 이동 궤적을 생성한다. 그리고 이동체 위치 매칭 모듈에서는 추출된 불확실성 영역 내의 추정 위치와 이력 데이터간의 차분을 토대로 위치를 교정한다.



(그림 4) 불확실성 처리기 구조도

5.2 주요 모듈 설계

본 절에서는 불확실성 처리기를 구성하는 주요 세부 모듈과 사용자 화면의 설계에 관하여 설명한다.

5.2.1 인접 시간 구간 검색 모듈(SearchInterval)

데이터베이스에 저장된 데이터 중 사용자가 입력한 질의 시점에 근접한 시간 구간 데이터 또는 시간 구간에서의 차량의 속도를 검색할 수 있다.

5.2.2 불확실한 위치 추정 모듈(Un_Position)

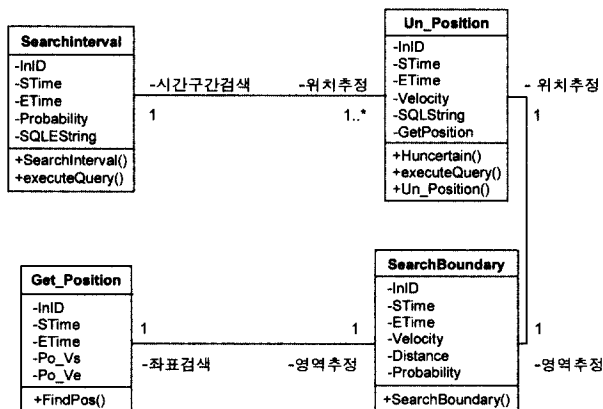
질의 시점에서의 차량의 불확실한 위치를 추정하기 위하여 검색한 시간 구간과 구간의 위치 좌표 및 선형 보간법을 사용한다. 이 좌표는 불확실한 영역의 중심 좌표가 된다. 불확실한 위치 추정모듈은 기존 MOMS에 존재하는 Positionattime연산자에 구현된 Huncertain 선형 보간 함수를 사용한다.

5.2.3 불확실한 영역 추출모듈(SearchBoundary)

불확실한 위치 추정모듈에서 추정된 위치 좌표를 중심으로 불확실한 영역을 구하는 모듈이다. 사용자가 입력한 불확실한 시간 영역과 구간 내의 차량의 속도를 사용하여 중심 좌표로부터 불확실한 시간 영역까지의 거리를 구할 수 있다. (거리=속도*시간) 구해진 거리는 불확실한 영역의 반경이 된다. 불확실한 영역은 이 반경을 중심으로 하는 원으로 가정할 수 있다.

5.2.4 불확실한 위치 채택모듈(Get_Position)

최종 불확실한 차량의 위치는 전날의 차량 위치 데이터 중에서 구해진 불확실한 영역에 존재하는 차량의 위치로 채택한다. 이 모듈에서는 전날 차량의 데이터 중 불확실한 영역에 존재하는 차량의 위치를 검색하여 위치가 검색 되지 않았을 경우 "검색 결과 없음", 하나의 차량 위치가 검색 될 경우 그 위치를, 그리고 둘 이상의 차량의 위치가 검색되었을 경우 전날 차량 데이터 중 입력 된 시점과 가장 가까운 시점의 차량데이터를 불확실한 위치로 채택한다.



(그림 5) 불확실성 처리기 클래스 다이어그램

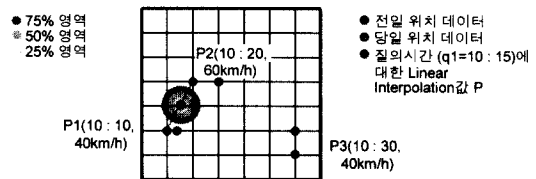
불확실성 처리기는 이동체 관리 엔진에 의해 호출되어 작업을 수행하며, 그 결과는 자체 윈도우 또는 상위 서브-시스템을 통해 결과를 반환한다. 불확실성 처리기는 이동체 위치정보 검색 메뉴의 각 항목으로부터 호출되는 서브 윈도우에서 확률 값을 설정한다. 또한 별도의 시스템 설정 메뉴를 통해 기타의 시스템 설정과 함께 불확실성 확률 값을 설정한다. 불확실성 처리기의 시스템 설정(기정의 확률 값 등)은 불확실성 확률 값을 0%로 하며, 불확실성 확률 값은 0~100% 사이의 값이다. 시스템 입출력은 다음과 같이 정의한다. 불확실성 처리기로 상위 서브-시스템에서 입력된 이동체 불확실성 질의(차량번호, 검색시간, 확률값)를 전달 받으며, 생성된 결과는 상위 서브-시스템에서 전자지도 출력 화면을 사용하여 출력한다.

5.3 위치 불확실성 처리 알고리즘

차량 위치 정보 불확실성 처리를 위한 기법은 최근접값 선택법(Nearest Value)와 선형보간법(Linear Interpolation)을 선택하여 적용할 수 있다. 최근접값 선택법은 저장된 이동체 위치 데이터 중에서 입력된 질의시간을 포함하는 두 개의 데이터에 대하여 하나를 반환하는 간단한 알고리즘을 가지며, 이에 비해 선택된 두 개의 데이터로부터 새로운 위치 데이터를 생성하는 선형보간법을 적용한 이동체 위치 불확실성 처리 알고리즘은 다음과 같다.

5.3.1 기본적인 처리 과정

- ① 사용자로부터 질의시간(t)과 확률값(pr)을 입력 받는다.
- ② 질의시간(t)을 포함하는 시간영역(t1, t2)에 대한 점(p1, p2)를 이동체 데이터베이스로부터 검색한다.
- ③ 선형보간법을 통해 질의시간(t)에 대한 위치기준점(p)을 계산한다. 이때, 위치기준점(p)은 p1, p2 선분을 대상으로 질의시간(t) 값에 비례하여 결정한다.
- ④ p1에 대응하는 속도와 입력 받은 확률 값(pr)을 기반으로 p의 불확실성 영역(UcR)을 표현한다.



(그림 6) 이동체 위치 불확실성 영역

5.3.2 추가적인 처리 과정

- ① 만일 질의시간(t)에 대응하는 p에 대한 이력 값(전일) p'이 있다면 p'으로 보정한다. 단 사용자가 입력한 확률 값에 의해 구한 불확실성 영역 내에 p'이 존재해야 한다.
- ② 만일 질의시간(t)에 대응하는 p의 불확실성 영역(UcR)에 p1 또는 p2가 포함되는 경우 p1 또는 p2로 수렴할 수 있다. 위의 (그림 7)에서 질의시간(t)가 10:12이고 확률 값(pr)이 50%인 경우에는 p는 p1으로 보정한다.

- ③ 위의 ①과 ②에 대하여 ②를 수행한 후 보정 값이 없는 경우 ①을 수행한다.
- ④ p의 값을 실제계의 의미로 부여하기 위해 Map matching을 수행할 수 있다. 이때, 기준은 불확실성 영역과 닿는 도로를 우선하며, 2개 이상의 도로가 닿는 경우에는 진행방향(p1과 p2의 위치 기준)을 고려하여 동서남북(EWSN)순으로 지정한다.

5.3.3 기타 사항

- ① 확률 값은 0%, 25%, 50%, 75%, 100%로 하며 불확실성 영역의 크기를 결정하는데 사용한다.
- ② 기정의값(Default value)은 0%으로 하며, 질의시간(t)이 데이터베이스에 저장된 시간 값 주기와 불일치하는 경우 “해당 값 없음”으로 반환한다.

6. 구현 및 평가

6.1 시스템 구현

6.1.1 불확실성 처리기 함수

불확실한 위치 처리기 알고리즘은 선형 보간 알고리즘과 전달 위치 중 입력한 불확실 영역에 포함되는 위치를 검색하는 알고리즘이 있다.

① 선형 보간 알고리즘

getInterval 함수는 선형 보간을 위하여 입력 된 시점이 포함된 시간 구간을 검색하는 함수이며, 입력된 시점의 정보를 입력 받아 입력 시점이 포함된 시간 구간을 검색한다. 시간 값은 선형 보간 함수에서 속도를 계산할 때 사용해야 하므로 분 단위로 변환한다.

Huncertain 함수는 데이터베이스에 저장 되지 않은 차량의 과거 위치정보를 추정하는 함수로서, (그림 7)에서와 같이 연산 처리 알고리즘은 데이터베이스로 관리되지 않는 질의 시점의 테이블 이름과 차량 식별자를 입력 값으로 받아서 데이터베이스에서 관리되지 않는 시점에서의 정보를 선형 보간식을 통하여 추출한다.

```

Funcion Huncertain
입력 → tname : 테이블 이름, id : 차량 ID, time : 질의 시점, interval :
      시간 구간
출력 → locx, locy : 질의 차량 위치 정보
public void Huncertain(String tname, int id, String time, int interval)
{
    rs ← time 시점이 포함된 하나의 튜플 정보를 이력 테이블에서 추출
    while(rs가 null이 아닌 경우) {
        fx ← 유효 시간 시작 시점에서 차량의 x좌표 정보를 rs로 부터 추출
        fy ← 유효 시간 시작 시점에서 차량의 y좌표 정보를 rs로 부터 추출
        tx ← 유효 시간 끝 시점에서 차량의 x좌표 정보를 rs로 부터 추출
        ty ← 유효 시간 끝 시점에서 차량의 y좌표 정보를 rs로 부터 추출
        lx ← (((tx-fx)/interval)×(time - cvf) + fx)공식을 이용하여 불확실
            한 객체의 x좌표 정보 추출
        ly ← (((ty-fy)/interval)×(time - cvf) + fy)공식을 이용하여 불확실
            한 객체의 y좌표 정보 추출
    }
}
    
```

(그림 7) Huncertain 함수 알고리즘

② 차량 이력정보 추출 알고리즘

IsInclude 함수는 하루 전달 동일 시점에서의 차량 위치 좌표가 사용자가 입력한 확률 값에 의하여 계산된 불확실한 영역 반경 안에 포함되는 지를 검사하는 함수이다. 선형 보간 된 위치 좌표로부터 불확실한 영역 반경이 포함되는 원 안에 입력 시점과 동일한 하루 전달의 시점에서의 위치 좌표가 포함되는 지를 검사한다. 특정 좌표가 원안에 포함되는 지를 검사하기 위하여 원 부등식 공식을 사용한다.

```

Funcion Huncertain
입력 → unPt : 선형 보간 위치 좌표, distance : 불확실한 영역 반경,
      beforePt : 동일 시점의 하루전 위치 좌표
출력 → result : 포함되는지 판단하기 위한 boolean값
boolean IsInclude(Point2D unPt, double distance, Point2D beforePt)
{
    // 원의 방정식 (x - a)^2 + (y - b)^2 ≤ r^2에 속하는 x, y는 원안에
    // 들어감
    (단 a, b는 중심 좌표 r은 distance)
    distxPt ← unPt.getX() - beforePt.getX(); // x - a
    distyPt ← unPt.getY() - beforePt.getY(); // y - b
    circle ← Math.pow(distxPt, 2) + Math.pow(distyPt, 2); // (x - a)^2
        + (y - b)^2
    radius ← Math.pow(distance, 2); // r^2
    if(circle ≤ radius) (x - a)^2 + (y - b)^2 // r^2일 경우 원안에 포함
        result = true ;
    else
        result = false ;
    return result ;
}
    
```

(그림 8) IsInclude 함수 알고리즘

또한 getDistance 함수는 하루 전 차량이 사용자가 입력한 확률 값 내부에 포함되는 지를 확인하기 위하여 확률 값을 불확실한 영역의 반지름으로 변환하는 함수이며, 사용자가 입력한 확률 값을 사용자 입력 시점의 시간 구간과 시간 구간에서의 속도 값을 사용하여 불확실한 영역의 반지름으로 변환하는 역할을 한다. 여기서 사용자가 입력한 확률 값은 데이터베이스에 입력된 시간 구간을 100%로 하였을 때의 확률 값이 된다.

6.1.2 구현 환경 및 시나리오

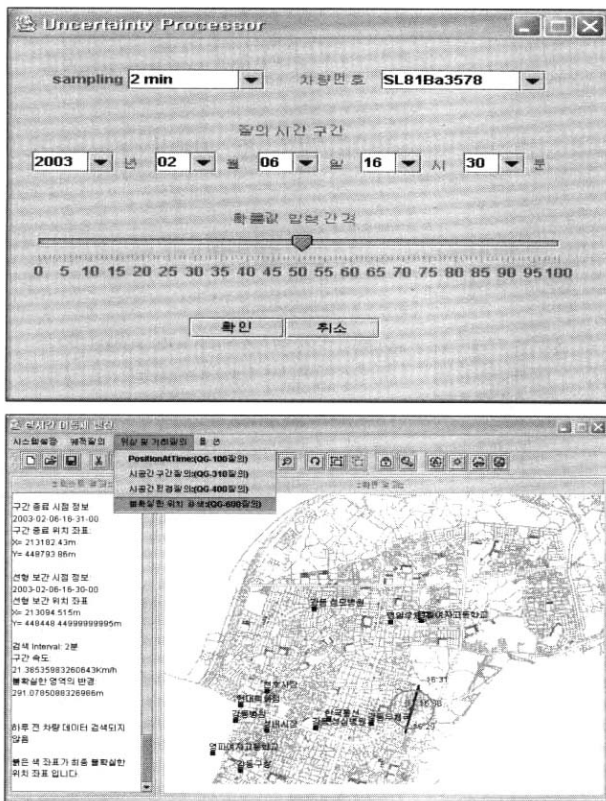
본 논문에서 제안한 차량 위치정보 불확실성 처리기는 Pentium IV PC에서 MS Windows XP기반의 Java SDK 1.3.1과 Oracle 9i를 토대로 구현되었으며, 실험 데이터 수집을 위해 위하여 다음과 같은 시나리오를 적용하고 관련 서버-시스템과의 패킷 통신을 수행하여 저장하였다.

차량 위치 데이터는 서울시 강동구와 성동구에 각각 GPS 및 CDMA장치를 부착한 가상의 택배차량(서울81바3578, 서울82아2468)을 대략 2시간(16:00~18:00) 동안 미리 설정된 계획 경로를 토대로 이동한 시간 및 이동 좌표를 사용하였다. 불확실한 위치 처리의 시험을 위하여 수신된 위치 데이터를 1분 또는 5분 간격으로 별도의 테이블에 저장하였다.

데이터베이스에 저장여부에 관계없이 차량의 위치에 대하여 본 논문에서 제안한 불확실성 처리기를 통해 추정하는 질의 화면과 그 결과는 (그림 9)와 같다. (그림 9)의 결과 화면은 해당 차량의 실제 위치 데이터는 16:29과 16:31에 저장되었으며, 입력된 질의시간 16:30의 위치 값은 선형보간법에 의해 계산된 추정값으로 전날 일치하는 차량의 이력 위치데이터가 없음을 의미한다.

즉, 차량 위치정보 불확실성 처리 모듈이 없는 경우 사용자가 임의의 검색시간을 질의하면 해당 차량의 위치 검색이 실패할 수 있으나, 본 논문에서 제시한 불확실성 처리기를 사용한 경우 이를 해결할 수 있다.

[질의] “2003년 2월 6일 16시 30분 시점에서 서울81바3578 차량의 위치를 50%확률로 예측하라”



(그림 9) 불확실성 질의 화면 및 처리결과

6.2 시스템 평가

본 논문에서 제안한 차량 위치정보 불확실성 처리기는 빈번하게 변경되는 차량 위치 데이터를 모두 데이터베이스에 저장하지 못함으로 인해 야기되는 검색 연산의 실패를 방지하고 선형보간법과 같이 간단한 연산을 통해 저장 대비 검색효율을 증대하는 장점을 가진다.

즉, 선형보간법을 통한 단순하고 적절한 위치 추정을 지원하여 저장된 위치 검색뿐만 아니라 이동 중의 추정위치에 대한 계산 기능을 통해 이동체 데이터베이스의 검색대비 저

장비용을 30~50% 감소시킬 수 있었다. 즉, 실험에서는 실제 차량 2대의 위치데이터를 1분에서 5분 간격으로 세분화하여 저장하였으며, 불확실성 처리기를 적용하여 저장되지 않은 시간대의 검색에 대한 검색실패를 최소화함과 동시에 지역의 크기와 특성에 따라서 기준 값인 1분 간격의 데이터를 2분에서 최대 5분까지도 지원하여 저장되는 데이터 분량을 감소시킬 수 있음을 확인하였다.

이를 위해서는 택배차량과 같이 복잡한 도로를 운행하는 차량의 이동에서 선형보간법에 의한 추정된 위치데이터는 도로가 아닌 위치에 존재할 수 있으며, 이를 해결하기 위해서는 전자지도 정보를 토대로 맵-매칭(Map-matching)하는 단계가 부가적으로 실행한다.

7. 결 론

정보통신의 발전으로 다양한 응용분야에서 요구하는 새로운 기술들이 개발되어 생활의 변화를 야기하고 있다. 대표적으로 위치를 추정하는 장치를 장착한 차량으로부터 전송 받은 위치 정보를 토대로 현재 차량의 위치 정보나 교통량의 흐름 등을 실시간으로 제공해 주는 차량 위치추적 시스템은 이미 택배를 포함한 e-Logistics 환경에서 점차 활용의 폭이 넓어지고 있다.

차량 위치 추적을 위한 데이터는 갱신 간격이 매우 짧고 빈번하게 발생하는 매우 거대한 분량을 갖는 일종의 시공간 데이터이다. 기존의 차량추적 관련 연구들은 방대한 데이터에 대한 효과적인 관리 방안을 갖지 못한 이유로 오직 가장 최근에 검출된 이동체 위치 데이터만을 관리하기 때문에, 데이터베이스에 저장되지 않는 과거의 불확실한 위치 추정 방안은 미약한 상태에 있다.

이 논문에서는 차량의 위치데이터에 이력 관리 기능을 추가한 이동체 관리 시스템의 핵심요소 중의 하나인 차량 위치정보 불확실성 처리 모형과 처리기 구현을 소개하였다. 세부적으로 모델링과 데이터베이스 및 시스템 구조, 불확실성 처리 모형과 시스템 구조를 설명하였으며, 가상의 택배 차량 운행 시나리오를 기반으로 제안한 시스템의 적용 예와 평가를 설명하였다.

본 논문에서 제안한 차량 위치 불확실성 처리기는 이동체 엔진을 포함하는 이동체 관리 시스템의 핵심 요소 중의 하나로서, 통합 위치데이터 인터페이스와 웹/모바일 프리젠테이션 인터페이스 등의 서브-시스템과의 통합과 이동체 라우팅 및 스케줄링 및 플랫폼과 같은 시스템간의 유기적인 연동을 통한 응용 시스템 구축 및 시범 운영 과정이 진행될 예정이다.

참 고 문 헌

[1] M. Erwig, R.H. Güting, M. Schneider and M. Vazirgiannis, "Spatio-Temporal Data Types : An Approach to Modeling and Querying Moving Objects in Databases," GeoInfor-

matica, Vol.3, No.3, pp.269-296, 1999.

[2] R. H. Gutting and et al., "A Foundation for Representing and Querying Moving Objects," ACM Transactions on Database Systems, Vol.25, No.1, pp.1-42, 2000.

[3] L. Forlizzi, R. H. Güting, E. Nardelli and M. Schneider, "A Data Model and Data Structures for Moving Objects Databases," Proc. of the ACM SIGMOD Conference, pp. 319-330, 2000.

[4] O. Wolfson, B. Xu, S. Chamberlain and L. Jiang, "Moving Objects Databases: Issues and Solutions," Proc. of the 10th International Conference on Scientific and Statistical Database Management, SSDBM'98, Capri, Italy, pp.111-122, 1998.

[5] P. Sistla, O. Wolfson, S. Chamberlain and S. Dao, "Modeling and Querying Moving Objects," Proc. of the 13th International Conference on Data Engineering, ICDE'97, Birmingham, UK, Apr., 1997.

[6] P. Sistla, O. Wolfson, S. Chamberlain and S. Dao, "Querying the Uncertain Position of Moving Objects," Springer Verlag Lecture Notes in Computer Science number 1399, pp.310-337, 1998.

[7] O. Wolfson, P. Sistla, B. Xu, J. Zhou, S. Chamberlain, N. Rishe and Y. Yesha, "Tracking Moving Objects Using Database Technology in DOMINO," Proc. of NGITS'99, The 4th Workshop on Next Generation Information Technologies and Systems, Zikhron-Yaakov, Israel, pp.112-119, 1999.

[8] O. Wolfson, S. Chamberlain, S. Dao, L. Jiang and G. Mendez, "Cost and Imprecision in Modeling the Position of Moving Objects," Proc. of the 14th International Conference on Data Engineering, ICDE'98, Orlando, FL, Feb., 1998.

[9] D. Pfoser, C. S. Jensen and Y. Theodoridis, "Novel Approaches in Query Processing for Moving Objects," Proc. of the VLDB Conference, pp.395-406, 2000.

[10] D. Pfoser and C. S. Jensen, "Capturing the Uncertainty of Moving Object Representations," Proc. of Advances in Spatial Databases, 6th International Symposium, SSD'99, pp.20-23, 1999.

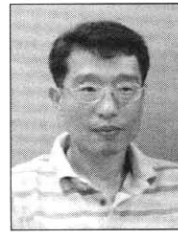
[11] S. Grumbach, P. Rigaux, M. Scholl and L. Segoufin, "The Design and Implementation of DEDALE," 1999.

[12] I. B. Oh, Y. A. Ahn, E. J. Lee, K. H. Ryu and H. G. Kim, "Prediction of Uncertain Moving Object Location," Proc. of International Conference on East-Asian Language Processing and Internet Information Technology, EALPIIT'02, pp.51-58, Jan., 2002.

[13] S. S. Park, Y. A. Ahn and K. H. Ryu, "Moving Objects Spatiotemporal Reasoning Model for Battlefield Analysis," Proc. of Military, Government and Aerospace Simulation part of ASTC'01, pp.108-113, 2001.

[14] M. H. Huh, Y. A. Ahn and K. H. Ryu, "Moving Object Location Change Function using Cubic Spline Interpolation," Proc. of the 6th World Multiconference on Systemics, Cybernetics and Informatics, SCI'02, pp.336-341, Computer Science I, 2002.

[15] Y. A. Ahn, D. H. Kim and K. H. Ryu "Design of A Moving Object Management System for Tracking Vehicle Location," The KIPS Transactions : Part D, Vol.9-D, No.5, Oct., 2002.



김진석

e-mail : kimjs@etri.re.kr

1982년 울산대학교 전자계산학과(학사)

1988년 동국대학교 대학원 전자계산학과
(공학석사)

2004년 충북대학교 대학원 전자계산학과
(이학박사)

1983년~현재 ETRI 우정기술연구센터 u-Post연구팀장/책임연구원

관심분야 : CSCW, 소프트웨어공학, 위치기반서비스, 물류정보시스템 등



김동호

e-mail : kdh@etri.re.kr

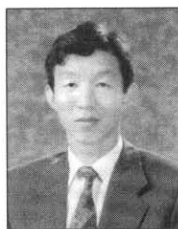
1993년 충북대학교 전자계산학과(학사)

1995년 충북대학교 대학원 전자계산학과
(이학석사)

1999년 충북대학교 대학원 전자계산학과
(이학박사)

1999년~현재 ETRI 우정기술연구센터 선임연구원

관심분야 : 시공간 데이터베이스, 이동체 기술, 지리정보시스템, 위치기반서비스, 물류응용정보시스템 등



류근호

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

1976년 숭실대학교 전산학과(이학사)

1980년 연세대학교 공업대학원 전산전공
(공학석사)

1988년 연세대학교 대학원 전산전공
(공학박사)

1976년~1986년 육군군수 지원사 전산실(ROTC 장교), 한국전자통신연구원(연구원), 한국방송통신대 전산학과(조교수) 근무

1989년~1991년 Univ. of Arizona Research Staff(TempIS 연구원, Temporal DB)

1986년~현재 충북대학교 전기전자컴퓨터공학부 교수

관심분야 : 시간 데이터베이스, 시공간 데이터베이스, Temporal GIS 및 지식기반 정보검색 시스템, 데이터 마이닝 및 데이터베이스 보안, 바이오 인포메틱스