

전장분석을 위한 이동 객체의 위치 예측 시스템

(A Location Prediction System for Moving Objects in Battlefield Analysis)

안 윤 애 [†] 류근호 ^{**} 조동래 ^{***}
(Yoon Ae Ahn) (Keun Ho Ryu) (Dong Lae Gho)

요약 전장분석을 위해서는 대상이 되는 적 부대에 대한 식별 및 이동 상황에 관한 정확한 정보가 요구된다. 그러나 통신 기술, 전파 방해, 전술 등의 이유로 모든 정보를 완벽하게 수집하는 것은 어렵다. 따라서 이미 수집된 이동 정보 및 도메인 지식을 이용한 미래 이동 상황의 예측 및 분석이 필요하다. 특히, 이동하는 부대들은 시간에 따라 위치 및 모양이 변하는 이동 객체의 특성을 가지므로, 전장분석을 위해서는 이동 객체를 관리하고 위치를 예측하는 기능이 필수적이다. 따라서 이 논문에서는 전장분석을 위한 이동 부대들의 위치를 예측하는 시스템을 제안한다. 이 시스템은 전장분석 응용 도메인에 적합한 미상의 부대 추론, 미확인 부대 추론, 주타격 방향을 예측하며, 데이터베이스에 저장되지 않은 이동 객체의 과거 및 미래의 위치를 추정한다.

키워드 : 전장분석 시스템, 이동 객체, 위치 추론

Abstract For the battlefield analysis, it is required to get correct information about the identification and moving status of target enemy units. However, it is difficult for us to collect all of the information perfectly, because of the technology of communications, jamming, and tactics. Therefore, we need a reasoning function that predicts and analyzes future moving status for target units by using collected moving information and domain knowledge. Especially, since the moving units have characteristics of moving objects, which change their position and shape over time, they require functions to manage and predict locations of moving objects. Therefore, in this paper, we propose a location prediction system of moving units for battlefield analysis. The proposed system not only predicts unknown units, unidentified units, and main strike directions to application domain for battlefield analysis, but also estimates the past or future locations of moving objects not stored in a database.

Key words : Battlefield Analysis System, Moving Objects, Location Inference

1. 서론

모의 전장에서 기본적으로 요구되는 지형, 기상, 부대의 이동 및 전술에 대한 데이터를 바탕으로 한 전장지역의 평가 및 분석을 전장분석(Battlefield Analysis)이라 한다. 전장분석을 위해서는 기본적으로 대상이 되는 적

부대의 위치 식별 및 이동 상황에 관한 정확한 정보가 요구된다. 그러나 모든 정보를 완벽하게 수집하는 것은 어렵기 때문에 이미 수집된 부대의 위치 변화 정보 및 관련 지식을 토대로 한 예측 및 분석이 필요하다. 그런데, 전장분석의 주요 대상이 되는 부대, 탱크, 차량 등은 시간에 따라 위치 및 모양이 변하는 이동 객체(Moving Objects)[1,2,3,4]의 특성을 가지고 있으며, 이는 시간 및 공간 속성을 모두 가지는 시공간 데이터의 일종에 포함된다. 이로 인해 적군의 대처와 시간의 변화에 따른 제약 등의 특수성을 고려할 때, 전장분석 시스템은 시공간 이동 객체를 관리하는 기능은 물론 전장분석 응용에 필요한 부대의 식별 및 위치를 적절히 추론하는 기능을 가져야 한다.

· 이 연구는 ADD의 연구비 지원과 한국과학재단 RRC(ICRC) 지원으로 수행되었음.

[†] 학생회원 : 충북대학교 전자계산학과

yeahn@dblab.chungbuk.ac.kr

^{**} 종신회원 : 충북대학교 전기전자 및 컴퓨터공학부 교수

khryu@dblab.chungbuk.ac.kr

^{***} 비회원 : 국방과학연구소 연구원

dicho@add.re.kr

논문접수 : 2000년 11월 7일

심사완료 : 2002년 7월 8일

지금까지 시공간 데이터의 추론은 주로 시간 추론과 공간 추론으로 구분되어 연구되었다. 시간 추론[5,6,7]은 시간 데이터가 가지는 여러 가지 특징들에 대한 추론으로 시간 개념을 형식화하고 지식의 시간적인 측면을 표현하고 추론한다. 공간 추론[5,8]은 공간 객체들 간의 위상, 방향, 거리 관계에 근거하여 여러 가지 공간 문제를 해결하기 위해 주로 연구되었다. 그러나 전장분석 응용에서는 미상의 부대를 식별하고, 미확인 부대를 추론하며, 주타격 방향을 예측할 수 있는 도메인에 한정된 추론 기능은 물론 일반적인 이동 객체의 위치를 예측하는 기능을 동시에 필요로 하기 때문에 기존의 시간 및 공간 추론 모델 및 알고리즘을 그대로 적용할 수 없다.

따라서 이 논문에서는 전장분석을 위한 도메인 추론 기능과 이동 객체의 위치 예측 기능을 동시에 제공하는 시스템을 설계 및 구현한다. 이 시스템은 데이터베이스, 이동 객체 연산기, 추론 엔진, 규칙베이스, GIS 도구로 구성된다. 데이터베이스에는 이동하는 적 부대의 위치 정보가 저장된다. 이동 객체 연산기는 데이터베이스에 저장된 위치 정보를 이용한 연산을 처리하며, 특히 이동 객체의 위치를 예측하는 연산도 함께 처리한다. 그리고 추론 엔진은 규칙베이스에 저장된 도메인 지식을 이용하여 전장분석에 필요한 미상의 부대, 미확인 부대, 주타격 방향을 추론한다.

제안 시스템은 이동 객체의 시간 및 공간 속성을 관리하기 위해 시공간 데이터베이스 형태로 데이터베이스를 구축하여 이동 객체들의 모든 이력정보를 관리할 수 있다. 또한, 이동 객체를 이용하는 특정 응용에 대한 추론을 수행하기 위해 도메인 규칙베이스와 시공간 연산을 이용한 도메인 추론엔진 모델을 제시한다. 아울러 위치 예측 연산을 통해 데이터베이스에 저장되지 않은 이동 객체의 과거 및 미래의 위치를 추정하는 특징을 가진다.

이 논문의 내용 전개는 다음과 같은 순서로 진행된다. 먼저, 2절에서는 이동 객체와 시간 및 공간 추론의 관련 연구를 검토하며, 기존의 연구가 전장분석을 위한 이동 객체의 위치 추론에 부적합한 이유를 분석한다. 3절에서는 전장분석을 위한 이동 객체의 위치 예측 시스템의 구성 및 기능, 그리고 위치 예측 연산 방법을 설명한다. 4절에서는 전장분석을 위한 미상의 부대, 미확인 부대, 주타격 방향 추론 수행 과정 및 이동 객체의 위치 예측 알고리즘을 기술한 후 구현 결과를 예시한다. 5절에서는 구현 시스템의 특성을 검토하고, 마지막으로 6절에서는 결론 및 향후 연구방향을 제시한다.

2. 관련연구

2.1 이동 객체

이동 객체란 시간에 따라 공간 객체의 모양이나 일반 속성이 변하는 기하 데이터 및 공간상의 객체가 이동하면서 위치 및 모양을 변화시키는 특징을 가지는 시공간 데이터이다. 이동 객체는 두 가지 기본 데이터 타입으로 이동 점과 이동 영역을 가진다[1,2,4]. 이동 점(Moving Points)은 공간에서 객체의 위치가 시간에 따라 변하는 특징을 가지며 사람, 동물, 차, 탱크 등과 같은 종류가 있다. 이동 영역(Moving Regions)은 위치의 변화뿐만 아니라 모양도 함께 변하며 영역의 성장 또는 축소가 가능하다. 병합, 폭풍, 암 등과 같은 데이터들이 이동 영역에 속한다.

이동 객체의 응용 시스템에 관한 대표적인 연구로는 먼저, 미국에서 개발된 DOMINO[9,10,11,12] 프로토타입이 있다. 이 시스템은 이동 객체의 현재 위치 정보와 속도 및 방향 정보를 이용하여 가까운 미래의 위치를 추정하는 기능을 제공하며, 미래 질의를 위한 FTL(Future Temporal Logic) 언어를 제시하였다. 그러나 이 시스템에서는 과거의 이력 정보를 저장하지 않기 때문에 과거 시점을 포함하는 이동 객체의 완전한 이동 경로를 관리할 수 없는 단점을 가진다. 또한 이동 객체의 미래 위치와 관련된 질의 처리에만 주로 초점이 맞추어졌기 때문에, 과거 정보의 저장 관리 및 이력 정보를 이용한 미래의 위치 추론 방법을 제시하지 못하고 있다. 이와 같은 이유로 과거 이력정보를 활용하여 미래의 이동 패턴을 추정하는 전장분석 응용 시스템에 적용하기에는 부적합한 특성을 가진다.

유럽의 CHOROCHRONOS 연구 컨소시움에서는 시공간 데이터베이스의 특수한 영역으로 이동 객체에 관한 연구가 수행되었으며 특히, 이동 객체의 데이터 모델링, 인덱싱, 불확실성 처리에 관한 연구[1,2,3,4,13,14]가 주로 진행되었다. 그리고 GPS 기반의 수송 관리 시스템과 멀티미디어 시스템을 적용 대상으로 선정하여 응용 시나리오를 제시하였다. 그러나 아직 이동 객체 데이터베이스를 활용한 응용 시스템의 모델 및 개발 사례는 제시되지 않은 상태이므로, 추론 기능을 갖는 응용 시스템에 적용할 수 있는 모델로는 부적합한 상태이다.

2.2 시간 및 공간 추론

시간 추론(Temporal Reasoning)에 대한 연구는 논리를 확장하여 시간을 표현할 수 있도록 하는 수학적 접근 방식[7]과 시간 추론 기능을 전문가 시스템에 적용한 접근 방식인 시간 추론 전문가 시스템이 있다. 시간 추론

전문가 시스템의 관련 연구 중 우주, 항공 분야에 적용된 몇 가지 예를 들면, 허블 천체 망원경의 제어를 위한 전문가 시스템인 WHEELS[15], 객체지향 전문가 시스템 도구인 REX[16], 그리고 사건기반 시간 추론 기법과 추론엔진 NEO를 이용하여 구현된 시간 추론 엔진인 NEO/Temporal[17]이 있다. 이 중 NEO/Temporal은 전장분석 응용과 유사한 군 작전 상황 판단 및 결정 시스템이다. 그러나 NEO/Temporal은 사전에 미리 입력된 경로로 적기가 침투하여 이동하는 상황에 대해서만 추론을 수행하므로, 동적으로 객체의 경로가 변하는 상황에 대해서는 적절한 위치를 추론하지 못하고 있다.

공간 추론(Spatial Reasoning)은 공간 데이터의 모델링에서 설정한 공간 도메인을 기반으로 공간 객체의 상호 관계를 다루기 위해 주로 위상 관계 추론과 방향 관계 추론[8,18]에 관해 연구되었다. 위상 관계 추론은 공간 데이터의 equal, meets, overlap, contains, disjoint, properly-contains, contained-in, properly-contained-in과 같은 위상 관계를 이용한 추론 방법이다. 방향 관계 추론은 공간에서 순서를 나타내는 공간 관계의 한 종류인 방향 관계에 대한 추론이다. 공간상의 두 점들 간의 방향 관계는 주로 동, 서, 남, 북, 북동, 북서, 남동, 남서와 같은 8개의 기본적인 관계가 이용된다. 지금까지 연구된 공간 객체의 위상 및 방향 추론의 개별적인 연구만으로는 공간상에서 이동하는 객체에 대한 현재, 과거, 미래의 위치 추론을 지원하지는 못한다.

시공간 추론(Spatiotemporal Reasoning)에 관한 연구에는 시간 및 공간 지식을 통합한 시공간 지식의 표현 구조 및 지리 정보 시스템에서의 시공간 추론 방법이 있다. Dutta는 [5]에서 시간 및 공간 개념의 통일된 표현 구조를 위상 제약사항으로 제시하였다. 지리 정보 시스템에서의 시공간 추론은 컴퓨터 과학적 시각에서의 시공간 추론, 시간 및 공간 인식, 사회 과학 분야에서의 시간 및 공간 작용 등에 관한 연구[19]가 수행되었다. 시공간 추론의 응용은 시공간 데이터베이스 및 전문가 시스템의 개발에 모두 관련이 있으며, 관련 응용에는 신진대사 이상 진단 시스템, 주식 투자 전문가 시스템, 게임 시뮬레이터 등이 있다. 이동 객체를 이용한 응용에는 항법 시스템, 워 게임 모델, 근접 전투 시뮬레이션 및 전장분석 시스템 등이 있다.

3. 전장분석을 위한 이동 객체의 위치 예측 시스템

이 논문에서는 모의 전장에 사용되는 부대, 탱크 등과 같은 이동 객체의 위치 정보와 관련된 지식만을 고려한

도메인 추론을 수행하고, 데이터베이스에 저장되지 않은 이동 객체의 과거 및 가까운 미래의 위치를 예측하는 시스템을 설계한다. 전장분석에 사용되는 이동 객체를 위한 위치 예측 시스템은 사용자 인터페이스, 데이터베이스, 이동 객체 연산기, 추론 엔진, 규칙베이스, GIS로 구성되며 그림 1과 같은 구조를 가진다.

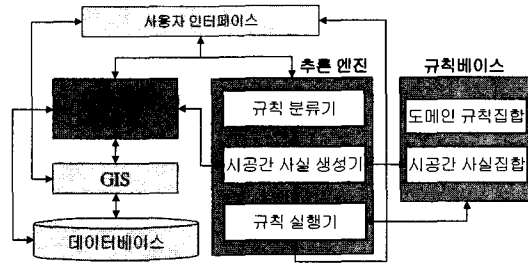


그림 1 전장분석을 위한 이동 객체의 위치 예측 시스템

그림 1에서 사용자 인터페이스는 GUI 환경을 제공한다. 데이터베이스는 이동 객체의 시간 및 공간 속성을 저장 및 관리하며, 기존의 상용 DBMS를 이용한다. 이동 객체 연산기는 데이터베이스에 저장된 이동 객체에 관한 검색 및 연산을 수행하며, 위치 예측 연산을 통해 데이터베이스에 저장되지 않은 이동 객체의 과거 및 미래의 위치를 예측한다. 추론 엔진은 데이터베이스에 저장된 이동 객체의 위치 정보와 규칙베이스에 저장된 도메인 지식, 그리고 이동 객체 연산기에 의해 처리된 시공간 사실정보들을 이용하여 전장분석을 위한 추론 기능을 수행한다. 규칙베이스는 추론 엔진의 수행에 필요한 도메인 규칙 및 시공간 사실을 저장한다. GIS 도구는 이동 객체 데이터의 공간 정보 및 공간 인덱싱 정보를 관리하며 지도정보를 관리한다.

3.1 전장분석을 위한 이동 객체

전장분석에 사용될 이동 객체는 이동 점으로만 가정한다. 이동 점으로 표현될 수 있는 이동 객체에는 부대, 탱크, 미사일, 사람 등이 있으며, 객체의 이동 변화에 대한 정보는 이산적인 모델로 표현한다. 이산 모델[1,2,4]에서는 이동 객체의 시공간 위치 변화 과정을 직선으로 표시한다. 이는 직선의 방정식을 이용하여 객체의 위치 변화를 선형적으로 근사시킴을 의미한다.

그림 2와 같이 이동 객체는 이차원 공간상의 x, y 좌표를 시간을 나타내는 t 축으로 확장하여 3차원으로 구성한다. t 축이 가질 수 있는 시간 값은 과거, 현재, 미래가 모두 가능하다. 그러나 실제 데이터베이스에 저장되는 값은 과거와 현재 값이고, 미래의 값은 특정 연산

에 의해 그 결과를 얻는다. 이동 객체가 가지는 공간 속성에는 (x, y) 위치 좌표, 시간 속성으로는 유효 시간(Valid Time)만을 가짐을 한정한다.

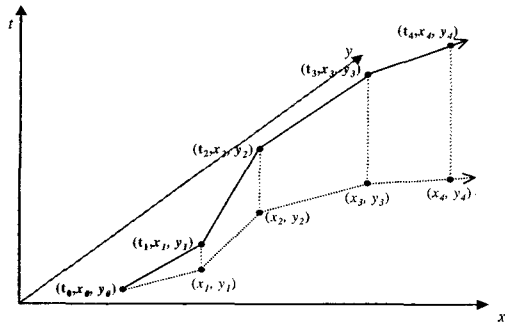


그림 2 이동 객체의 위치 변화

3.2 데이터베이스

전장분석을 위한 데이터베이스는 초기 부대 정보 릴레이션과 이동 정보 릴레이션, 미상의 부대 릴레이션으로 구성된다. 초기 부대 정보 릴레이션에는 처음으로 관측된 적 부대들의 일반 속성, 위치 좌표, 그리고 실제 관측된 날짜인 유효시간이 기록된다. 부대의 초기 정보는 공간 인덱싱을 처리하기 위해 GIS에 저장된다. 이 논문에서 사용한 GIS 도구는 Geowin[20]이며, Java 언어로 구현되었다.

표 1은 부대의 초기 정보를 저장하기 위한 초기 부대 정보 릴레이션의 구조로 객체 식별자(Id), 코드명(Code), 명칭(Name), 유효 시작 시간(VTs), 유효 종료 시간

(VTe), VTs 시점의 X좌표(X_vts), VTe 시점의 X좌표(X_vte), VTs 시점의 Y좌표(Y_vts), VTe 시점의 Y좌표(Y_vte), 부대별 구분자(Type), 연대정보(Reg)로 구성된다. 이동 정보 릴레이션은 초기 부대 정보 릴레이션에 저장된 부대들의 모든 이동 과정을 기록하는 이력 릴레이션으로 부대들의 위치 좌표와 이동 날짜가 기록된다. 이동 정보 릴레이션은 데이터베이스에 저장되며, 이 논문에서는 Oracle DBMS를 사용한다.

표 2와 같이 이동 정보 릴레이션의 구조는 이동 객체의 코드명(Code), 유효 시작 시간(VTs), 유효 종료 시간(VTe), VTs 시점의 X좌표(X_vts), VTe 시점의 X좌표(X_vte), VTs 시점의 Y좌표(Y_vts), VTe 시점의 Y좌표(Y_vte)로 구성된다. 미상의 부대 릴레이션에는 부대 식별 정보가 존재하지 않고, 단지 부대의 위치 좌표와 시간값 만이 획득된 정보를 저장한다.

표 3의 미상의 부대 릴레이션에는 부대의 식별 정보가 존재하지 않으므로 이동 객체의 Code가 저장되지 않고, 유효 시작 시간(VTs), 유효 종료 시간(VTe), VTs 시점의 X좌표(X_vts), VTe 시점의 X좌표(X_vte), VTs 시점의 Y좌표(Y_vts), VTe 시점의 Y좌표(Y_vte) 만이 저장된다.

3.3 이동 객체 연산기

이동 객체 연산기는 데이터베이스에 저장된 이동 객체들을 이용한 연산을 처리하는 부분이다. 이동 객체 연산기에서 처리되는 연산 유형은 일반적인 이동 객체 연산자와 데이터베이스에 저장되지 않은 과거 및 미래의 위치 예측 연산이 있다.

3.3.1 이동 객체의 일반 연산

표 1 초기 부대 정보 릴레이션(INIT_IPB)

Id	Code	Name	VTs	VTe	X_vts	X_vte	Y_vts	Y_vte	Type	Reg
키값	객체의 코드	객체의 명칭	유효시간 (시작)	유효시간 (종료)	VTs 시점의 X좌표	VTe 시점의 X좌표	VTs 시점의 Y좌표	VTe 시점의 Y좌표	부대별 구분자	연대정보
int	int	string	string	string	float	float	float	float	int	string

표 2 이동 정보 릴레이션(HISTORY_IPB)

Code	VTs	VTe	X_vts	X_vte	Y_vts	Y_vte
객체의 코드	유효시간 (시작)	유효시간 (종료)	VTs 시점의 X좌표	VTe 시점의 X좌표	VTs 시점의 Y좌표	VTe 시점의 Y좌표
int	string	string	float	float	float	float

표 3 미상의 부대 릴레이션(UNCERTAIN_IPB)

VTs	VTe	X_vts	X_vte	Y_vts	Y_vte
유효시간 (시작)	유효시간 (종료)	VTs 시점의 X좌표	VTe 시점의 X좌표	VTs 시점의 Y좌표	VTe 시점의 Y좌표
string	string	float	float	float	float

전장분석을 위해 다음과 같은 이동 객체 연산자를 정의한다. 이동 객체의 일반 속성에 대한 검색은 기존의 상용 DBMS가 제공하는 기능을 그대로 사용하므로 별도로 정의하지 않는다. 표 4는 기존의 이동 객체 연구[3]에서 제시된 연산자들을 토대로 기본 연산만을 정의한 것이다.

표 4의 이동 객체 일반 연산 이외에 전장분석을 위해 필요한 연산들을 표 5와 같이 추가로 정의하며, 이를 시공간 사실 생성 연산자라 한다. 시공간 사실 생성 연산은 추론 엔진의 규칙 실행에 필요한 사실들을 생성한다.

표 5의 연산에서 T_Near, T_Far, T_VeryFar는 사람에 따라서 느끼는 거리 개념의 정도가 서로 달라서 하나의 수치 값으로 표현이 불가능한 공간 특성을 가진다[17,21]. 따라서 실제 시스템에서는 T_Near, T_Far, T_VeryFar에 대해 정량화된 값의 범위를 미리 할당하여 처리한다.

3.3.2 과거의 위치 예측 연산

과거의 위치 예측 함수는 $PCF : t_k \rightarrow (x_{t_k}, y_{t_k})$ 형태를 띠며, 임의의 시간 t_k 를 입력 값으로 받아서 그 결과로 (x_{t_k}, y_{t_k}) 의 위치 좌표를 반환한다. 이 때 t_k 의 범위는 $t_0 \leq t_k \leq now$ 이다. t_0 는 초기 부대정보 릴레이션에 저장된 최초의 시간이고, now 는 이동 정보 릴레이션에 저장된 가장 최근의 종료 시점이다. 함수 PCF 는 두 점을 지나는 직선의 방정식을 이용하며, $x = PCF(t_k)$ 와 $y = PCF(t_k)$ 로 구성된다. $x = PCF(t_k)$ 함수는 임의의 시간 구간 $T_{x_i} = [t_{x_i}, t_{x_{i+1}}]$ 에 대해 $t_{x_i} \leq t_k \leq t_{x_{i+1}}$ 의 조건을 만족하는 과거의 임의의 시점 t_k 에 대한 x 좌표 값을 구하는 함수이다. 두 점 (t_{x_i}, x_i) 와 $(t_{x_{i+1}}, x_{i+1})$ 에 대한 함수

$PCF(t_k)$ 는 식 (1)과 같다.

$$x = \frac{x_{i+1} - x_i}{t_{x_{i+1}} - t_{x_i}}(t_k - t_{x_i}) + x_i \quad (1)$$

$y = PCF(t_k)$ 함수는 임의의 시간 구간 $T_{y_i} = [t_{y_i}, t_{y_{i+1}}]$ 에 대해 $t_{y_i} \leq t_k \leq t_{y_{i+1}}$ 의 조건을 만족하는 과거의 임의의 시점 t_k 에 대한 y 좌표 값을 구하는 함수이다. 두 점 (t_{y_i}, y_i) 와 $(t_{y_{i+1}}, y_{i+1})$ 에 대한 $PCF(t_k)$ 는 식 (2)와 같다.

$$y = \frac{y_{i+1} - y_i}{t_{y_{i+1}} - t_{y_i}}(t_k - t_{y_i}) + y_i \quad (2)$$

3.3.3 미래의 위치 예측 연산

미래의 위치 예측 함수는 $PPF : t_f \rightarrow (x_{t_f}, y_{t_f})$ 형태를 띠며, 임의의 시간 t_f 를 입력 값으로 받아서 그 결과로 (x_{t_f}, y_{t_f}) 의 위치 좌표를 반환한다. 이 때 t_f 의 범위는 $now < t_f$ 이다. now 는 이동정보 릴레이션에 저장된 가장 최근의 시점이므로, t_f 는 항상 미래의 임의의 특정 시점을 나타낸다. 이동 객체의 3차원 이력 집합을 $P = \{(t_i, x_i, y_i)\}_{i=1}^n$ 이라고 할 때, P 를 각각 x 와 y 에 대해 2차원 집합으로 분해하면 $P_x = \{p_x(t_{x_i}, x_i)\}_{i=1}^n$ 와 $P_y = \{p_y(t_{y_i}, y_i)\}_{i=1}^n$ 이 된다.

이제 $x = PPF(t_f)$ 와 $y = PPF(t_f)$ 두 개의 함수를 구하여 미래의 위치를 계산한다. 미래의 특정 시점에 대한 x 좌표 값은 집합 P_x 를 대상으로, 식 (3)과 같이 구한다.

$$\begin{aligned} x &= (d_{mx} * t_{mf}) + x_n \quad (3) \\ &= ((d_{tx}/t_{tx}) * (t_f - t_{x_n})) + x_n \\ &= (\sum_{i=2}^n (x_n - x_{n-1}) / \sum_{i=2}^n (t_{x_n} - t_{x_{i-1}})) * (t_f - t_{x_n}) + x_n \end{aligned}$$

미래의 특정 시점에 대한 y 좌표 값은 집합 P_y 를 대

표 4 이동 객체의 일반 연산

종류	입력	출력	연산식
MDistance	$O_A(x,y), O_B(x,y)$ $\langle VTs, VTe \rangle$	$\langle VTs, VTe \rangle$ 동안, O_A 와 O_B 의 거리(float)	$\sqrt{(Oy(B) - Oy(A))^2 + (Ox(B) - Ox(A))^2}$
Trajectory	Oid_{O_A} , $\langle VTs, VTe \rangle$	$\langle VTs, VTe \rangle$ 동안, O_A 의 이동 궤적(line)	$Y - Oy(VTe) = \frac{Oy(VTe) - Oy(VTs)}{Ox(VTe) - Ox(VTs)}(X - Ox(VTe))$
Length	Oid_{O_A}	O_A 의 총 이동거리(float)	$\sum_{i=1}^{n-1} Dist(Trajectory(O_{A_i}))$

표 5 전장분석을 위한 시공간 사실 생성 연산

종류	기능
NearestLoad(A,t1,t2)	유효시간 $\langle t1, t2 \rangle$ 동안에 객체 A와 가장 가까운 도로 검색
OppositeLoad(A,t1,t2)	유효시간 $\langle t1, t2 \rangle$ 동안에 객체 A와 반대 방향의 도로 검색
T_MDirection(P1,P2)	P1(x1,y1) 위치에서 P2(x2,y2) 위치로 이동한 방향
T_Near(A,t1,t2)	유효시간 $\langle t1, t2 \rangle$ 동안에 객체 A와 Near 관계에 있는 모든 객체를 검색
T_Far(A,t1,t2)	유효시간 $\langle t1, t2 \rangle$ 동안에 객체 A와 Far 관계에 있는 모든 객체를 검색
T_VeryFar(A,t1,t2)	유효시간 $\langle t1, t2 \rangle$ 동안에 객체 A와 VeryFar 관계의 모든 객체를 검색

상으로 하여 다음의 식 (4)와 같다.

$$\begin{aligned}
 y &= (d_{my} * t_{mj}) + y_n & (4) \\
 &= ((d_y / t_y) * (t_f - t_{y_n})) + y_n \\
 &= (\sum_{i=2}^n (y_n - y_{n-1}) / \sum_{i=2}^n (t_{y_n} - t_{y_{n-1}})) * (t_f - t_{y_n}) + y_n
 \end{aligned}$$

식 (3), (4)에서 t_f 는 $PPF(t_f)$ 의 입력 값으로 미래의 특정 시점을 의미한다. d_{mx} 는 x 축으로의 평균 이동거리, d_{my} 는 y 축으로의 평균 이동거리를 나타낸다. t_{mj} 는 현재부터 미래의 특정 시점까지의 이동시간, d_x 는 x 축으로의 총 이동거리, d_y 는 y 축으로의 총 이동거리, t_x 는 x 축으로의 총 이동시간, 그리고 t_y 는 y 축으로의 총 이동시간을 의미한다.

3.4 GIS 도구

이 논문에서는 GIS 도구로 Geowin 공간 관리 시스템을 사용한다. Geowin[20]의 데이터 저장은 모두 세 부분으로 구성된다. 객체의 비공간 속성 정보를 저장하는 속성 정보 테이블(ADT)과, 인덱싱 정보를 저장하는 공간 인덱싱 정보 테이블(SIT), 그리고 공간 정보를 저장하는 공간 정보 테이블(SOT)로 구성된다. 이 중에서 사용자가 임의로 원하는 속성정보를 생성 및 변경할 수 있는 테이블은 속성 테이블로 테이블 이름에 “ADT”라는 부가적인 이름이 첨부된다. Geowin의 공간 인덱싱 정보 테이블은 공간 객체들에 대한 R*-Tree 인덱스를 생성하여 저장한다.

3.5 도메인 규칙베이스

규칙베이스(Rule Base)는 도메인 규칙 집합과 사실 집합으로 구성된다. 도메인 규칙 집합에는 미상의 부대, 미확인 부대, 주타격 방향 등의 추론에 사용되는 도메인 규칙들이 저장된다. 규칙[22]은 그림 3과 같이 조건부(If~부분)와 행동부(Then~부분)로 이루어진다.

```

Rule_100() {
  if (element_of(_Mechanized_unit, _Motion_Table)
    && exist_on(_Mechanized_unit, _N_L)
    && opposite_direction(_N_L, _S_L)) {
    is_determined(Main_strike, _N_L);
    is_determined(Sub_strike, _S_L);
    display(Main_strike);
    display(Sub_strike);
  }
}
    
```

그림 3 Java 언어로 작성된 규칙 예

그림 3은 Java 언어의 if문으로 표현된 규칙의 예로서, 부대의 향후 주타격 방향을 결정하는 규칙이다. Rule_100은 규칙 번호를 나타내며, _Mechanized_unit, _Motion_Table, _N_L, _S_L, Main_strike, Sub_strike

등은 변수를 나타낸다. 그리고 element_of, exist_on, opposite_direction, is_determined, display 등은 규칙 실행 메소드이다.

3.6 추론 엔진의 실행 모델

추론 엔진은 규칙 분류기, 시공간 사실 생성기, 규칙 실행기로 구성되며, 전장분석을 위한 도메인 추론 기능을 수행한다. 그림 4는 추론엔진의 실행 모델을 나타낸 것이다.

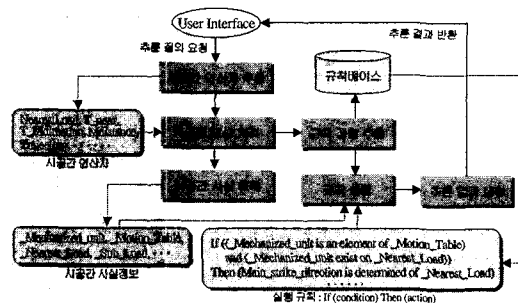


그림 4 도메인 규칙을 이용한 추론 모델

그림 4의 추론 엔진 모델은 다음과 같은 단계로 실행된다.

- 단계 1 : 시공간 연산자 추출
 - 입력된 추론 질의를 처리하는 데 필요한 시공간 연산자 추출
 - 질의 유형에 따라 서로 다른 시공간 연산 수행
 - 예 : NearestLoad, T_near, T_Mdirection, Mdistance, Trajectory 등
- 단계 2 : 시공간 연산 처리
 - 단계 1에서 추출한 시공간 연산 처리 후 시공간 사실정보 획득
 - 시공간 사실정보는 규칙 실행 시 조건부의 값으로 할당
 - 예 : _Mechanized_unit, _Motion_Table, _Nearest_Load, _Sub_Load 등
- 단계 3 : 규칙 집합 추출
 - 규칙베이스로부터 입력된 질의 유형에 필요한 규칙 집합 추출
 - 규칙 집합은 If (조건부) Then (행동부)로 구성되는 규칙의 모임
- 단계 4 : 규칙 실행
 - 단계 2에서 생성된 시공간 사실정보와 단계 3에서 획득한 규칙 집합 이용
 - 작업 메모리에서 규칙 실행 후 결과를 생성

4. 구현

전장분석을 위한 이동 객체의 위치 예측 시스템은 클라이언트/서버 구조로 구현되었다. 클라이언트는

Window NT 4.0 환경에서 Java 1.2로 구현하였고, 서버는 UNIX 환경에서 Oracle 7과 Geowin 공간관리 도구를 활용하였다. 시스템의 기능별 구현 범위는 전장분석을 위한 추론 질의와 일반 질의로 구분된다. 전장분석을 위한 추론 질의에는 미상의 부대 추론, 미확인 부대 추론, 주타격 방향 추론(이동중, 집결배치), 부대의 이동 위치 추론, 부대의 이동시간 추론 기능이 있다. 일반 질의 유형에는 부대의 이동경로 추적, 일별 부대정보 조회, 근접 부대 검색, 버퍼 검색 등이 있으며, 사용자 인터페이스를 통한 이동 정보의 입력 기능도 제공한다.

구현된 전체 시스템의 모듈은 크게 이동 객체 연산기와 추론 엔진으로 구분된다. 이동 객체 연산기에서 구현된 연산자의 범위는 3.3절에서 제시한 연산자와 사용자 질의 수행을 위해서 시스템 내부적으로 처리되는 내부 연산자들이 포함된다. 추론 엔진은 규칙 분류 모듈, 사실 정보 생성 모듈, 규칙 실행 모듈로 구현되었으며, 이 논문에서 제시한 전장분석 도메인 추론 기능에 한정된 처리 기능을 가진다. 규칙 실행을 위한 도메인 규칙 집합은 각 추론 질의별로 서로 다르게 구성되었다. 이 절에서는 전체 구현 범위 중에서 이 논문에서 초점을 맞추고 있는 전장분석 추론 기능별 실행 과정과 이동 객체의 위치 예측 연산 처리 과정을 기술한 후 구현 결과를 예시한다.

4.1 전장분석 도메인 추론 기능

4.1.1 미상의 부대 추론

미상의 부대 추론은 관측된 이동 정보 중에서 날짜와 위치는 정확하지만 부대의 명칭이나 소속 정보가 부정확한 경우, 해당 부대의 명칭과 소속정보를 추론하는 기능이다. 규칙이 실행되어 추론되는 과정은 다음과 같다.

- 단계 1 : 미상의 부대 후보 집합 생성(S1)
 - 초기 객체 정보 중에서 이력 정보가 없는 객체들을 선택
 - 즉, 이동 정보가 존재하지 않는 모든 부대를 미상의 부대 후보 집합으로 생성
- 단계 2 : 미상의 부대 후보의 예상 위치 집합 생성(S2)
 - 집합 S1의 모든 원소에 대해 미상의 부대 후보의 예상 위치 집합 S2를 생성
 - 이동 위치 예측 연산 활용
- 단계 3 : 이동 정보가 관측된 미상의 부대 집합 생성(S3)
 - 미상의 부대 릴레이션에서 질의 입력 날짜에 해당되는 위치 정보를 모두 추출
 - 이동 정보가 관측된 미상의 부대 집합 S3를 생성
- 단계 4 : 가장 유사한 미상의 부대 후보 추출 및 중간 결과 집합 생성(S4)
 - 집합 S3의 각 원소에 대해, 집합 S2의 원소들과의 거리 계산
 - 계산된 거리가 가장 가까운 부대 선택 후 중간 결과 집합 S4에 저장
- 단계 5 : 중간 결과 집합의 원소와 동일한 연대에 소속된 부대 집합 생성(S5)

- 집합 S4의 각 원소에 대해 동일한 연대에 소속된 부대 집합 S5를 생성
- 단계 6 : 거리 관계 조건 검사 및 최종 결과 집합 생성(S6)
 - 집합 S4의 각 원소에 대해, 집합 S5의 원소들과의 거리 관계 조건 검사
 - 거리 관계 조건 예) 동일한 연대에 속한 부대들간의 거리 (D) : $60 \leq D \leq 90$
 - S4의 원소 중 거리 관계 조건을 만족하는 원소만을 최종 결과 집합 S6에 저장

4.1.2 미확인 부대 추론

미확인 부대 추론은 획득된 이동 정보가 전혀 없는 부대의 명칭과 위치를 추론하는 기능이다. 미확인 부대를 추론하기 위한 규칙 실행 단계는 다음과 같다.

- 단계 1 : 미확인 부대 후보 집합 생성(U1)
 - U1 = 미상의 부대 후보 집합(S1) - 미상의 부대 추론 결과 집합(S6)
- 단계 2 : 미확인 부대 후보 집합의 예상 위치 추론(U2)
 - 집합 U1의 모든 원소에 대해 미확인 부대 후보의 예상 위치 추론 집합 U2를 생성
 - 이동 위치 예측 연산 활용
- 단계 3 : 동일한 연대에 소속된 부대 집합 생성(U3)
 - 집합 U2의 각 원소에 대해, 동일한 연대에 소속된 부대 집합 U3를 생성
- 단계 4 : 거리 관계 조건 검사 및 최종 결과 집합 생성(U4)
 - 집합 U2의 각 원소에 대해, 집합 U3의 원소들과의 거리 관계 조건 검사
 - U2의 원소 중 거리 관계 조건을 만족하는 원소만을 최종 결과 집합 U4에 저장

4.1.3 주타격 방향 추론

주타격 방향 추론은 부대의 이동 정보를 이용하여 향후 부대들의 이동 방향을 예측하는 기능이다. 주타격 방향 추론은 이동 중과 집결배치 후로 구분되는데, 이동 중의 주타격 방향 추론은 집결배치 후 주타격 방향 추론의 단순화이므로 집결배치 후 주타격 방향 추론을 수행하는 단계만 설명한다. 주타격 방향 추론 수행 단계는 다음과 같다.

- 단계 1 : 기계화 부대와 가장 인접한 도로 정보 생성(L)
 - 이동정보 릴레이션에서 질의 날짜의 유효 시간에 포함되는 기계화 부대 정보 추출
 - 기계화 부대와 가장 인접한 도로 탐색 및 정보 추출
- 단계 2 : 전방 연대가 그룹핑된 형태에 관한 정보 생성(F)
 - 이동정보 릴레이션에서 질의 날짜의 유효 시간에 포함되는 전방 연대의 정보 추출
 - 전방 연대의 위치 정보를 토대로, 구성 형태가 삼각형인지 일선형인지 구분
- 단계 3 : 후방 연대가 치우친 방향 정보 생성(B)
 - 이동정보 릴레이션에서 질의 날짜의 유효 시간에 포함되는 후방 연대의 정보 추출
 - 후방 연대의 위치 정보를 토대로, 가장 인접한 도로정보 추출
- 단계 4 : 가중치 비교 및 결과 생성
 - 생성된 L, F, B 정보에 대해 질의에서 입력받은 가중치 비교
 - 가중치가 가장 큰 정보와 가장 인접한 도로를 기준으로

주타격 방향 결정

-가중치가 모두 동일할 경우에는 세 가지 조건 각각에 대한 주타격 방향을 모두 제시

4.2 이동 객체의 위치 예측 연산

이동 객체의 과거 및 미래 위치 예측 연산은 미상의 부대 및 미확인 부대 추론 규칙에 사용되는 시공간 사실 집합을 생성한다. 과거 및 미래의 위치 추정 연산 알고리즘은 3.2절에서 제시한 데이터베이스를 이용하여 3.3절에서 기술한 수식을 처리하는 알고리즘이다.

4.2.1 과거의 위치 추정 연산

데이터베이스에 저장되지 않은 과거의 특정 시점의 이동 위치 예측 알고리즘은 3.3.2절의 위치 변화 함수 $x = PCF(t_k)$ 와 $y = PCF(t_k)$ 를 구하는 방법과 동일하다.

```

Algorithm PCF(oid_O, t_k)
입력 값 => oid_O : 임의의 이동 객체 O의 식별자, t_k : 과거의 특정 시점
출력 값 => x_t_k : 시점 t_k 시 O의 x 좌표 값, y_t_k : 시점 t_k 시 O의 y 좌표 값
INIT_IPB : 초기 부대 정보 릴레이션, HISTORY_IPB : 이동 정보 릴레이션
Begin
  INIT_IPB에서 조건 Id=oid_O 를 만족하는 튜플의 Code를 검색
  If (검색 결과가 null이 아니면) Then
    code ← Code
    HISTORY_IPB에서 조건 Code=code and VTs<=t_k and t_k<=VTe 를 만족하는 튜플검색
    RS ← 검색 결과
    If (RS가 null이 아니면) Then
      x_t_k ← (x_vt_e - x_vt_s) / (vt_e - vt_s) * (t_k - vt_e) + x_vt_e
      y_t_k ← (y_vt_e - y_vt_s) / (vt_e - vt_s) * (t_k - vt_e) + y_vt_e
    Else
      x_t_k ← 오류 값, y_t_k ← 오류 값
    Return x_t_k and y_t_k //과거의 t_k 시점의 (x,y)좌표값 반환
  End
  
```

그림 5 과거의 위치 추정 알고리즘

그림 5의 알고리즘 PCF(oid_O, t_k)에서 입력 값은 임의의 이동 객체 O의 식별자인 oid_O와 과거의 특정 시점 t_k이고, 출력값은 t_k 시점의 (x,y) 좌표인 x_t_k와 y_t_k이다. 여기에서 x_t_k의 값은 3.3.2절의 식 (1)을 구하는 방법과 동일하고, y_t_k의 값은 식 (2)와 동일하다.

4.2.2 미래의 위치 추정 연산

이동 객체가 미래의 특정 시점에 존재할 위치 예측 알고리즘은 3.3.3절의 위치 예측 함수 $x = PPF(t_f)$ 와 $y = PPF(t_f)$ 를 구하는 방법과 동일하다.

그림 6의 알고리즘 PPF(oid_O, t_f)에서 입력 값은 임의의 이동 객체 O의 식별자인 oid_O와 미래의 특정

```

Algorithm PPF(oid_O, t_f)
입력 값 => oid_O : 임의의 이동 객체 O의 식별자, t_f : 미래의 특정 시점
출력 값 => x_t_f : 시점 t_f 시 O의 x 좌표 값, y_t_f : 시점 t_f 시 y 좌표 값
INIT_IPB : 초기 부대정보 릴레이션, HISTORY_IPB : 이동정보 릴레이션
N : 미래의 위치 좌표 추정에 사용될 최근 이력 튜플 수
Begin
  If (INIT_IPB에서 Id = oid_O인 객체가 존재) Then
    임시변수 code에 객체의 Code값을 할당
    If (HISTORY_IPB에서 Code = code and Time = t_f 튜플 검색) Then
      x_t_f에 X값을 할당, y_t_f에 Y값을 할당
    Else
      current ← HISTORY_IPB에서 가장 마지막에 저장된 튜플의 time 값
      current_x ← current 시점의 x 좌표 값
      current_y ← current 시점의 y 좌표 값
      HISTORY_IPB에서 current 이전의 N개의 튜플을 검색
      검색된 N개의 튜플의 time, x, y값을 t_i, x_i, y_i 배열에 저장
      d_t_x, d_t_y, t_t_x에 초기치 0을 할당
      DO i=0, N-2
        d_x ← x_{i+1} - x_i //1번째 튜플의 x축 이동거리
        d_t_x ← d_t_x + d_x //x축 이동거리의 합
        d_y ← y_{i+1} - y_i //1번째 튜플의 y축 이동거리
        d_t_y ← d_t_y + d_y //y축 이동거리의 합
        t_x ← t_{i+1} - t_i //1번째 튜플의 이동시간
        t_t_x ← t_t_x + t_x //이동시간의 합
      End DO
      d_m_x ← d_t_x / t_t_x //x축 평균 이동거리
      d_m_y ← d_t_y / t_t_x //y축 평균 이동거리
      t_m_f ← t_f - current //향후 이동시간
      x_t_f ← (d_m_x * t_m_f) + current_x //미래의 x좌표
      y_t_f ← (d_m_y * t_m_f) + current_y //미래의 y좌표
    Else
      x_t_f ← 오류 값, y_t_f ← 오류 값
    Return x_t_f and y_t_f //미래의 t_f 시점의 (x,y) 좌표 값
  End
  
```

그림 6 미래의 위치 추정 알고리즘

시점 t_f이고, 출력값은 t_f 시점의 (x,y) 좌표인 x_t_f와 y_t_f이다. 여기에서 x_t_f의 값은 3.3.3절의 식 (3)을 구하는 방법과 동일하고, y_t_f의 값은 식 (4)와 동일하다.

4.3 적용 시나리오 및 구현 결과

구현 시스템의 위치 예측 기능 실험을 위해 모의 전장 시나리오를 구성하였다. 이 시나리오는 육군 1개 사단이 이동하고 있을 때 현재 상황을 분석하여 예상 공격 방향을 미리 예측하여 대응 전략을 세우도록 하는 것이다. 먼저, 초기 부대정보 테이블에는 1개 사단의 20개 대대를 이동 객체로 가정하여 2000년 5월 1일에 관측된 정보를 입력하였다. 이후 20개의 대대에 대해 10일간의 이동 정보를 발생시켰다. 전체 200회의 이동 정보 중에서 정확한 정보는 80%(160회)이고 나머지 20%(40회)는 부정확한 정보로 가정한다. 이와 같은 시나리오를 대상으로 다음에 몇 가지 질의 수행 결과의 예를 보인다.

그림 7은 '2000/05/09~2000/05/10'의 유효시간에 해당되는 미상의 부대를 추론한 결과 화면이다. 점선 타원의 내부는 '2000/05/09~2000/05/10'까지의 부대 위치 정보를 나타낸다. 타원 안의 작은 실선 원은 미상의 부대로 추론된 부대의 위치정보 및 부대 명칭을 나타낸다.

그림 8은 '2000/05/13~2000/05/14'의 유효시간에 해당되는 미확인 부대를 추론한 결과 화면이다. 이동 객체

데이터베이스에 저장된 이동정보는 '2000/05/12'까지만 존재하며, '2000/05/13' 이후의 유효 시간에 해당되는 모든 객체들의 위치는 미확인 부대로 추정된 결과이다.

그림 9는 '2000/05/11~2000/05/12'까지 부대가 집결 배치 한 후 향후 이동할 주타격 방향을 추론한 결과 화면이다. 원 내부에는 질의에 입력된 유효시간에 해당되는 객체들의 위치를 나타낸 것이고 화살표로 표시된 부분은 '2000/05/13' 이후 부대들이 이동할 주타격 방향을 추론한 결과이다.

그림 10은 '11-보대'가 향후 '2000/05/13~2000/05/15'까지 이동할 이동 위치를 추론한 결과이다. 사각형 내부는 '11-보대'가 '2000/05/01~2000/05/11'까지 이동한 경로를 나타내며, 원의 내부는 향후 '2000/05/13~2000/05/15'까지의 예상 이동경로를 나타낸다.

그림 11은 '22-보대'가 현재 위치에서 (620,520) 위치까지 이동하는데 걸리는 시간을 추론한 결과이다. 사각형 내부는 '22-보대'가 '2000/05/01~2000/05/11'까지 이동한 경로를 나타내며, 원의 내부는 향후 (620, 520) 지점까지 도달하는데 소요되는 예상 시간 및 위치를 나타낸다.

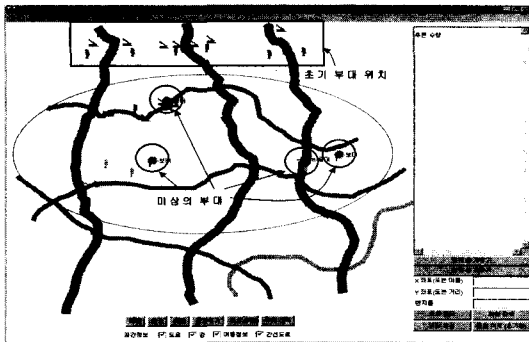


그림 7 미상의 부대 추론

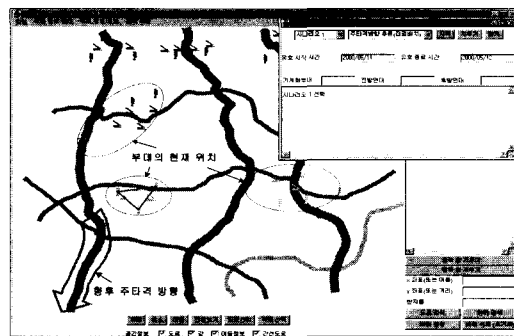


그림 9 주타격 방향 추론

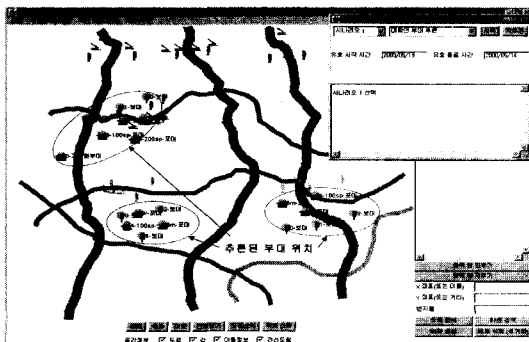


그림 8 미확인 부대 추론

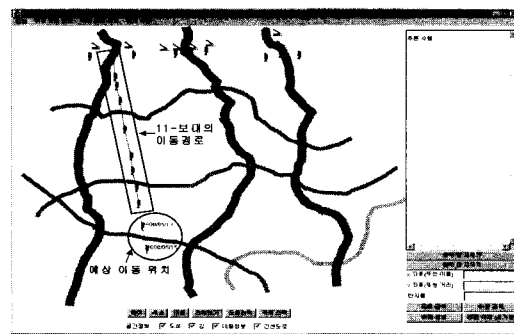


그림 10 이동 위치 추론

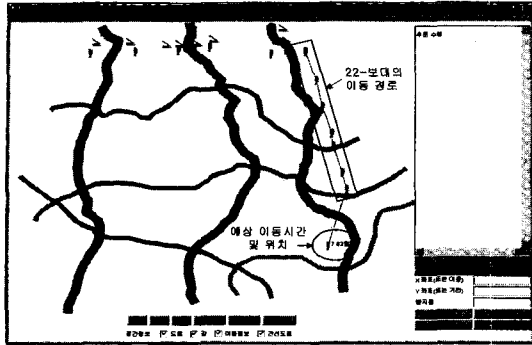


그림 11 이동 시간 추론

5. 구현 시스템의 특성 분석

구현 시스템의 특성 분석 및 고찰을 위해 시스템의 추론 기능별 일별 질의 응답률과 이동 위치 예측 연산을 이용한 위치 추정 결과의 오차를 분석하였다.

5.1 추론 기능별 질의 응답률

일별 질의 응답률은 시스템에 질의한 일별 전체 질의 횟수에 대한 시스템의 응답횟수를 백분율로 나타낸 것이다. 실험을 위해 사용된 질의 유형은 다섯 가지이며, 모두 500개의 질의를 수행한 결과 그림 12와 같은 결과를 얻었다. 그림 12는 미상의 부대 추론(질의1), 미확인 부대 추론(질의2), 주타격 방향 추론(질의3), 부대의 이동 위치 추론(질의4), 부대의 이동시간 추론(질의5)의 다섯 가지 기능에 대해 이동정보가 획득된 기간별로 1일에서 10일까지 일별 50개의 질의를 작성하여 실행한 실험 결과이다. 이 중에서 하나의 추론 기능에 대해 일별 10회의 서로 다른 유효시간을 할당하여 질의를 작성하였다.

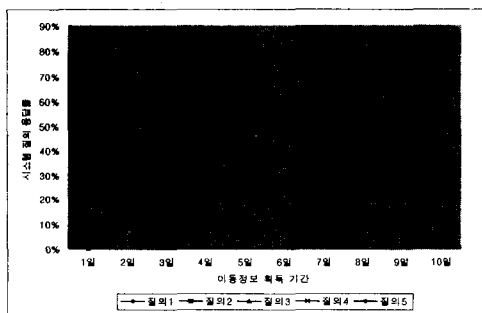


그림 12 질의 응답률

질의 응답률 그래프에서 모든 질의에 대해 대체적으로 이동정보의 획득 기간이 늘어날수록 시스템의 질의 응답률이 증가하는 경향을 볼 수 있다. 이 중에서도 특

히, 질의 1, 2, 3의 응답률이 질의 4, 5의 응답률보다 비교적 높은 분포를 나타냄을 알 수 있다. 그 이유는 질의 1, 2, 3은 전장분석이라는 특정 응용에 한정된 추론 기능이기 때문이다. 질의 1, 2, 3은 관련된 도메인 규칙을 활용하므로 이력 정보가 존재하지 않아도 시스템이 결과를 반환할 수 있지만, 질의 4와 5는 객체의 이동 위치와 시간에 관련된 질의이므로 이력정보가 없는 경우에는 그 결과를 얻지 못하게 된다.

5.2 위치 예측 연산의 추정 오차

이동 위치 예측 연산을 이용한 위치 추정 결과의 오차 분석을 위해 그림 13과 같이 가상 실험에서 사용된 지도를 이용하여 세 구간의 도로에서 이동하는 객체의 과거 및 미래의 위치 추정 실험을 하였다.

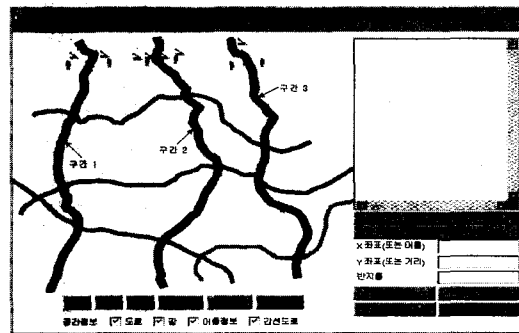


그림 13 위치 추정 실험 구간

먼저, 실험 대상이 되는 이동경로의 완전한 이력 위치 정보가 필요하므로, 구간 1, 2, 3에서 이동하는 객체의 완전한 이동경로를 선정하여 각각의 이동 시점과 위치 좌표에 관한 정보를 미리 저장하였다. 한 구간의 완전한 이동경로 집합은 $S = \{(t_i, x_i, y_i)\}_{i=0}^n$ 이고, i 는 정수이다. 이 때 (t_i, x_i, y_i) 는 i 시점의 위치 좌표를 나타낸다. 실험을 위해 구간 1, 2, 3에서 생성한 완전한 이동경로의 집합은 $S_1 = \{(t_i, x_i, y_i)\}_{i=0}^{124}$, $S_2 = \{(t_i, x_i, y_i)\}_{i=0}^{126}$, $S_3 = \{(t_i, x_i, y_i)\}_{i=0}^{127}$ 이다.

5.2.1 과거 위치 추정 결과의 오차

과거 위치 추정 결과의 실험을 위해 세 구간의 완전한 이동경로 집합 S_1, S_2, S_3 에 대해 위치 샘플 간격 즉, 시간의 주기를 나타내는 i 값을 2에서 10까지 변화시키며 새로운 위치 샘플을 발생시켰다. 이렇게 생성된 위치 정보는 $S_{1,k}, S_{2,k}, S_{3,k}$ 가 되고 $k=2, \dots, 10$ 까지 변화하며, 구간 1, 2, 3에 대해 모두 9개의 서로 다른 집합으로 저장하였다. 생성된 위치 정보를 이용하여 완전

한 이동경로 집합에 저장된 모든 과거 시점에 대해 과거 위치 추정 연산을 수행하였다. 과거 위치 추정 연산 후 얻어진 이동 경로 집합은 $P_{1,k}$, $P_{2,k}$, $P_{3,k}$, $k=2, \dots, 10$, 구간 1, 2, 3에 대해 각각 9개의 과거 위치 추정 결과 집합이 생성되었다. 이렇게 생성된 결과의 오차 계산은 위치 추정 결과 집합과 원래의 완전한 이동경로 집합에 저장된 위치 값과의 절대오차를 구하여 계산하였다. 각 구간 별 과거 위치 추정 오차는 그림 14와 같이 나타났다.

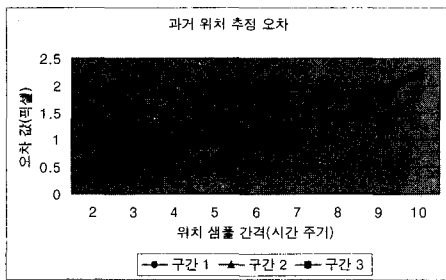


그림 14 과거 위치 추정 결과의 오차

그림 14의 그래프에서 x축은 시간 주기에 따른 위치 샘플의 간격을 나타내고, y축은 각 구간별 위치 샘플 간격에 따른 위치 추정 오차 값을 나타낸다. 위치 샘플 간격은 위치 좌표를 획득하는 일정한 시간의 주기를 의미한다. 따라서 위치 샘플의 간격이 적을수록 더욱 많은 이동정보가 데이터베이스에 저장된다. 실험 결과를 보면 모든 구간에서 공통적으로 위치 샘플 간격이 커짐에 따라 과거 위치 추정 오차 값이 커짐을 알 수 있었다. 이는 실제 데이터베이스에 저장된 위치 정보량이 많아질수록 과거 위치 추정 결과의 정확도가 향상됨을 의미한다.

5.2.2 미래 위치 추정 결과의 오차

미래 위치 추정 연산의 결과에 대한 오차 분석을 위해, 4.2.2절의 그림 6의 미래 위치 추정 알고리즘에서 사용된 과거의 이력 튜플 수 N의 값을 2 ~ 10까지 변화시켜가며 실험하였다. 미래의 위치 추정에 사용된 임의의 미래 시점은 완전한 이동경로 집합 S_1 , S_2 , S_3 에 저장된 시점과 동일하다. 미래 위치 추정 연산 후 얻어진 이동 경로 집합은 $F_{1,k}$, $F_{2,k}$, $F_{3,k}$, $k=2, \dots, 10$, 구간 1, 2, 3에 대해 각각 9개의 미래 위치 추정 결과 집합이 생성되었다. 이렇게 생성된 결과의 오차 계산은 위치 추정 결과 집합과 원래의 완전한 이동경로 집합에 저장된 위치 값과의 절대오차를 구하여 계산하였으며, 각 구간 별 미래 위치 추정 오차는 그림 15와 같이 나타났다.

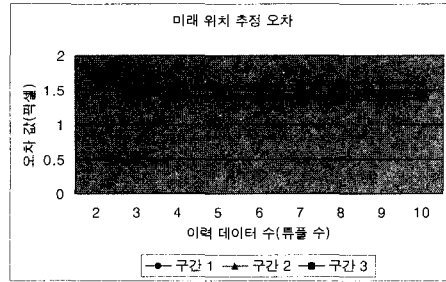


그림 15 미래 위치 추정 결과의 오차

그림 15에서 x축은 미래 위치 추정에 사용된 이력 튜플의 수를 나타내고, y축은 각 구간별 위치 추정 오차 값을 나타낸다. 실험 결과를 보면 미래 위치 추정에 사용된 이력 튜플의 수가 3과 4일 때 가장 오차가 적게 발생됨을 알 수 있다. 그래프의 전체적인 변화 추이를 보면 미래 위치 추정에 사용된 과거 이력 튜플의 수가 증가할수록 오차 값이 적어지다가, 어느 지점이 지나면 더 이상 오차 값의 변화가 많지 않음을 알 수 있다. 이는 미래 위치 추정 연산에 사용된 이력 튜플의 수가 많다고 해서 반드시 오차가 적어지는 것이 아니라, 각 이동 경로의 형태에 따라 최적의 N 값이 달라짐을 의미한다. 따라서 특정 응용분야에 따라 적절한 N값의 사용이 요구된다.

6. 결론

전장분석 응용에서는 적 부대의 이동 상황에 따라 아군의 적절한 대응전략이 필요하며, 이를 위해서는 적 부대의 이동 상황 및 위치에 관한 예측이 필수적이다. 이 논문에서는 전장분석에 적용되는 부대, 탱크 등의 이동 객체를 바탕으로 도메인에 종속적인 추론 및 위치 예측 시스템을 구현하였다. 그리고 가상 시나리오를 이용한 모의 실험을 통해 구현 시스템이 항법 시스템, 워 게임 모델, 근접 전투 시뮬레이션 등의 유사한 응용 시스템 개발에 적용 가능함을 알 수 있었다.

구현 시스템은 이동 객체의 시간 및 공간 속성을 관리하기 위해 시공간 데이터베이스 형태로 데이터베이스를 구축하여 이동 객체들의 모든 이력정보를 관리한다. 아울러, 이동 객체를 이용하는 특정 응용에 대한 추론을 수행하기 위해 도메인 규칙베이스와 시공간 연산을 이용한 도메인 추론엔진 모델을 제시하였으며, 위치 예측 연산을 통해 데이터베이스에 저장되지 않은 이동 객체의 과거 및 미래의 위치를 추정할 수 있는 특징을 가진다.

시스템의 질의 응답률을 분석한 결과 이동 객체의 위

치 예측 시스템에서는 축적된 이동 정보량이 구현 시스템의 질의 응답률에 영향을 미침을 알 수 있었으며, 적용 분야의 특성에 적합한 이동 정보 획득 기간의 결정이 중요한 변수가 됨을 알았다. 또한, 위치 예측 연산 결과의 오차 추정 실험을 통해 과거 위치 추정 연산은 이력 정보의 양이 많을수록 정확한 결과를 얻을 수 있었다. 미래의 위치 추정 연산은 이동경로의 형태에 따라 적합한 이력 튜플의 수가 달라짐을 알았다.

이동 객체들은 시간에 따라 속성, 위치, 모양 등이 변화되므로 샘플링 에러 또는 측정 에러로 인한 불확실성 [13,14]을 항상 가지게 된다. 추론 시스템은 추론 엔진에 사용되는 데이터나 지식이 모두 정확하다 하더라도 추론된 결과의 정확성을 보장하지는 못한다. 이와 같은 이유로 이동 객체 추론에 적용되는 시공간 데이터의 불확실성을 줄일 수 있는 방법 및 실제 추론 과정에 적용하여 결과의 정확성을 높이는 연구가 진행 중이다. 그리고 대용량 이동 객체 데이터의 효율적인 처리를 위한 인덱스 구조에 대한 연구가 진행될 것이다.

참고 문헌

- [1] M.Erwig, R.H.Güting, M.Schneider, and M. Vazirgiannis, "Abstract and Discrete Modeling of Spatio-Temporal Data Types," *Chorochronos Technical Report, CH-98-14*, 1998.
- [2] M.Erwig, R.H.Güting, M.Schneider, and M. Vazirgiannis, "Spatio-Temporal Data Types : An Approach to Modeling and Querying Moving Objects in Databases," *GeoInformatica Vol. 3, No.3*, 1999, pp.269-296.
- [3] R.H.Güting, M.Böhlen, M.Erwig, C.S.Jensen, N. Lorentzos, M.Schneider, and M. Vazirgiannis, "A Foundation for Representing and Querying Moving Objects," *ACM Transactions on Database Systems, Vol. 25, No. 1, Mar. 2000*, pp. 1-42.
- [4] L.Forlizzi, R.H.Güting, E.Nardelli, and M.Schneider, "A Data Model and Data Structures for Moving Objects Databases," *Proceedings of the ACM SIGMOD Conference, Dallas, Texas, May 2000*, pp. 319-330.
- [5] S.Dutta, "Topological Constraints : A Representational Framework For Approximate Spatial And Temporal Reasoning," *Second International Symposium on Large Spatial Databases, SSD'91*, 1991.
- [6] A.Montanari and B.Pernici, "Chapter 21 : Temporal Reasoning," *Temporal Databases*, Benjamin Cummings Publisher, 1993, pp. 534-562.
- [7] L.Vila, "A Survey on Temporal Reasoning in Artificial Intelligence," *AI communication, Vol. 7, No. 1, Mar. 1994*.
- [8] M.Grigni, D.Papadias, and C.Papadimitriou, "Topological Inference," *Proceedings of the International Joint Conference of Artificial Intelligence (IJCAI)*, AAAI Press, Montreal, Canada, 1995.
- [9] P.Sistla, O.Wolfson, S.Chamberlain, and S.Dao, "Modeling and Querying Moving Objects," *Proceedings of the Thirteenth International Conference on Data Engineering(ICDE13)*, Birmingham, UK, Apr. 1997.
- [10] O.Wolfson, S.Chamberlain, S.Dao, and L.Jiang, "Location Management in Moving Objects Databases," *Proceedings of the Second International Workshop on Satellite-Based Information Services (WOSBIS'97)*, Budapest, Hungary, Oct. 1997.
- [11] O.Wolfson, B.Xu, S.Chamberlain, and L.Jiang, "Moving Objects Databases: Issues and Solutions," *Proceedings of the 10th International Conference on Scientific and Statistical Database Management (SSDBM98)*, Capri, Italy, Jul. 1998, pp. 111-122.
- [12] O.Wolfson, P.Sistla, B.Xu, J.Zhou, S.Chamberlain, N.Rishe, and Y.Yesha, "Tracking Moving Objects Using Database Technology in DOMINO," *Proceedings of NGITS'99, The Fourth Workshop on Next Generation Information Technologies and Systems, Zikhron-Yaakov, Israel, Jul. 1999*, pp. 112-119.
- [13] D.Pfoser and C.S.Jensen, "Capturing the Uncertainty of Moving-Object Representations," *Advances in Spatial Databases, 6th International Symposium, SSD'99, Hong Kong, China, Jul. 1999*.
- [14] D.Pfoser and N.Tryfona, "Fuzziness and Uncertainty in Spatiotemporal Applications," *Chorochronos Technical Report, CH-00-4, Feb. 2000*.
- [15] W.Perkins and A.Austin, "Adding Temporal Reasoning to Expert System Building Environments," *IEEE Expert, Vol. 5, No. 1, Feb. 1990*, pp. 23-30.
- [16] B.Prasad, T.Parraju, G.Uma and P.Umarani, "An Expert System Shell for Aerospace Applications," *IEEE Expert, Vol. 9, No. 4, Aug. 1994*, pp. 56-64.
- [17] 이종현, 이민석, 우영운, 박충식, 김재희, "사건 기반 시간 추론 기법", *전자공학회 논문지, 제34권, 제5호, 1997년*, pp. 337-346.
- [18] J.Hong, M.Egenhofer, and A.Frank, "On the Robustness of Qualitative Distance and Direction-Reasoning," *Autocarto 12, Charlotte, NC, D. Peuquet(Ed.)*, Feb. 1995.
- [19] M.Egenhofer, "Spatial and Temporal Reasoning in Geographic Information Systems," *Oxford University press*, 1998.
- [20] Geostchnology, "Geowin System," <http://www.>

geowin.net.

- [21] J.Sharma, D.Flewelling, and M.Egenhofer, "A Qualitative Spatial Reasoner," Sixth International Symposium on Spatial Data Handling, Edinburgh, Scotland, Sep. 1994, pp. 665-681.
- [22] S.Dutta, "Knowledge Processing & Applied Artificial Intelligence," Butterworth Heineman Publisher, 1994, pp. 31-54.



안 윤 애

1993년 한남대학교 전자계산공학과(공학사). 1996년 충북대학교 대학원 전자계산학과(이학석사). 1999년 ~ 현재 충북대학교 대학원 전자계산학과 박사과정. 관심분야는 이동 객체 데이터베이스, 시공간 데이터베이스, 지리정보 시스템, 지식

기반 정보검색 시스템

류 근 호

정보과학회논문지 : 컴퓨팅의 실제
제 8 권 제 1 호 참조



조 동 래

1973년 연세대학교 전자공학과(공학사). 1979년 연세대학교 전자공학과(공학석사). 1991년 연세대학교 전자공학과(공학박사). 1978년 ~ 현재 국방과학연구소 책임연구원. 관심분야는 정보융합, 지식기반 시스템, 시공간 추론