

가상 환경에서 태양 복사와 바람 현상의 논리적 시뮬레이션 방법

조진영[†] · 박종희^{††}

요약

지금까지 가상 환경을 무대로 하는 게임 등에서 자연 현상의 처리는 시각적인 사실성에 치우친 그래픽 처리 위주로 연구되어져 왔다. 그러나 인과관계(causality)를 고려하지 않은 자연 현상들의 발생과 법칙에 기반하지 않은 현상들의 전개는 논리적 사실성을 감소시켜 사용자들의 몰입감을 떨어뜨린다. 본 논문에서는 자연 현상 중에서 태양복사(solar radiation)와 바람에 초점을 맞추어서, 이들의 다양하고 예측하기 어려운 발생과 전개과정을 시뮬레이션 하는 것을 목적으로 한다. 이를 위해 ontology에 기반한 객체들의 구조를 살펴보고, 이들을 바탕으로 가상 환경을 구성하였을 때 가상 환경의 데이터를 관리하는 방안을 제시한다. 그리고 각 자연 현상 이벤트에 대한 과정을 단계별로 상세히 다루고, 나아가 객체가 정성적 변화를 하였을 때 상황을 처리하는 방법을 제시한다.

Simulation of solar radiation and wind events in the virtual environments

JinYoung Cho[†] · Jong-Hee Park^{††}

ABSTRACT

Computer simulation of natural phenomena has been inclined to graphic processing for visual reality. This negligence of cosmic causalities in their occurrence and natural laws in their development should lead to limited degree of immersion to the users. We attempt to develop a logical framework for authentic simulation of diverse, unpredictable occurrence and development of natural phenomena (such as solar radiation and wind) based on their associated inherent laws and principles. To this end we structure the relevant objects organized in an ontology and propose a data management method. Then we describe our simulation method for the natural phenomena as delimited in phases and present modeling techniques for qualitative changes of physical objects due to their factors' values beyond normal ranges.

키워드 : 태양 복사(Solar Radiation), 자연 현상(Natural Phenomena), 가상 환경(Virtual Environment)

1. 서론

가상현실은 삼차원 컴퓨터 그래픽, 높은 해상도의 디스플레이, 인터랙티브 디바이스 등을 사용함으로써 사용자가 실제 세계에 있는 것처럼 느낄 수 있도록 해주는 컴퓨터가 만들어낸 세계이다. 그 기술들은 컴퓨터 그래픽스, CAD, GIS, 멀티미디어, 게임 등 다양한 분야에서 적용되어 왔다[1]. 또한 최근에는 컴퓨터 기반 교육용 시스템의 연구가 고조되면서, 교육의 효과를 높이기 위한 가상현실의 연구가 많이 진행되고 있다. 특히, 현실에서 위험성을 많이 가지고 있거나, 교육 환경 조성의 비용이 큰 화재진압 훈련, 비행 훈련, 원격조정 차량 훈련 등과 같은 분야에서 교육 시스템의 이용은 매우 효과적이다[2]. 이런 교육 시스템들은 사용자들이 주어진 도메인에서 현실에서와 유사한 경험을 할 수 있

도록 해주는 것이 중요하다.

지금까지 가상현실 연구는 사용자에게 현실감을 주기 위해 주로 시각, 청각적인 요소에 치중해왔다[3]. 이러한 요소들은 어느 정도 사용자들에게 감각적인(즉 시각과 청각) 사실성을 제공해 줄 수는 있다. 하지만 규칙(rule) 기반의 인과 관계가 배제된 이벤트들의 발생 및 전개, 정해진 환경의 변화는 사용자들이 느끼는 가상 환경에 대한 논리적 사실성을 감소시킨다. 논리적 사실성은 가상 환경에서 발생하는 물리적이고 자연적인 모든 현상들이 실제 세계에서와 비슷하게 발생하고 전개되는 것을 말한다. 논리적 사실성을 높이기 위해서는 규칙에 의해 이벤트가 발생하고 전개되어야 하고, 또한 그렇게 함으로써 다양하고 예측이 어려운 상황이 전개되어진다.

지금까지 다양한 상황을 전개하기 위하여 Relation-based model[4], 인과관계(causality)에 기반한 이벤트 전개모델[5, 6] 등이 소개되어져 왔다. Relation-based model은 관계(re-lations)를 미리 설정하기 때문에 적용할 수 있는 범위가 좁

※ 본 연구는 한국과학재단 목적기초연구(R05-2003-000-11941-0)지원으로 수행되었음.

† 정 회 원 : 드림소프트 주식회사 주임 연구원

†† 정 회 원 : 경북대학교 전자공학과 교수

논문접수 : 2003년 5월 16일, 심사완료 : 2003년 11월 27일

고, 다양성을 미리 제한한다는 단점이 있다. 인과관계 기반의 이벤트 전개모델은 이벤트의 발생 및 전개가 상황에 의존적이면서도 또한 상황을 변화시키므로 가상현실에서 자연 현상들을 구현하는데 적합하다고 할 수 있다. 그러나 가상현실에 존재하는 객체의 정성적 변화(qualitative changes)와 같은 극단적 상황(extreme cases)에서의 예측치 못한 이벤트를 처리하는 데는 미흡하다는 단점이 있다.

본 연구에서는 실제적 상황의 정확한 모사를 목표로 Physical Simulation과 달리 상황 전개에 다양성을 최대한 경험하게 하는데 주요점을 둔다. 따라서 자연현상을 최대한 자연의 법칙에 따라 시뮬레이션 함으로써 미리 설계된 경로를 통해서만 사건이 발생하고 전개되는 한계를 극복하고자 한다. 이를 위해 먼저 자연의 구성요소인 물리적 객체들을 단순히 수동적인 존재가 아닌 능동적 개체 즉 내부적 및 외부적 자극에 적절히 반응하는 기능을 갖는 개체들로 설계한다. 한편 객체들 사이의 관계를 여러 가지 측면에서 설정함으로써 최대한 다양화한다. 이와 관련하여 환경을 객체들의 단순한 배경이 아닌 객체들 사이의 상대적 관계로 설정한다. 동일한 상황들에 대해서 시간적, 공간적인 좌표축에 따라 인간의 다양한 필요에 따른 수준별 시각들을 제공할 수 있도록 구조화한다. 이를 위해 기존의 객체들에 대한 계층구조와 함께 관계들과 법칙들에 대해서도 계층구조를 도입한다. 그리고 비정형적(arbitrary) 요소들은 기본적으로 전형적인 값(typical value)과 변이(deviation)의 함으로 근사화한다. 그리고 관련된 규칙, 즉 법칙과 제약조건들을 최대한 수식화(formulation) 한다. 상황의 다양성을 최대화하는 또 다른 방안으로서, 상황의 정상적인 변동 범위 내에서의 정량적 변화뿐이 아닌 관련된 객체들 및 관계들에 정성적인 변화를 초래하는 상황들도 수용 가능하도록 시뮬레이션 모델을 설계한다.

본 논문에서는 자연 현상 중에서 태양 복사(solar radiation)와 바람 현상의 시뮬레이션을 예로 시뮬레이션 모델을 개발한다. 이들 현상의 전개는 인과관계 모델에 기반하여 설계되어지고, 또한 객체의 정성적 변화에 의한 예측할 수 없는 상황을 어떻게 처리할 것인가를 간단히 소개하겠다. 이벤트를 다루기전에 물리적 객체들의 구성에 대해서 소개하고, 이들에 의해 구성되어진 가상 환경에 대해서 다루도록 하겠다.

본 논문의 2장에서는 여기서 다루어질 주제들과 관련 있는 연구를 살펴보고, 3장에서는 객체의 구조에 대해서 다루고 이를 바탕으로 가상 환경의 정보를 가지는 Area tree를 소개할 것이다. 4장에서는 자연 현상 이벤트들의 진행과정(procedure) 및 각 단계에서 이용되어진 수식을 소개하고, 객체의 정성적 변화에 대해서 다루도록 하겠다. 5장에서는 앞에서 다루어진 것들을 실제로 구현하고 테스트하는 과정을 보여주고, 6장에서는 결론 및 앞으로의 연구 방향을 제시한다.

2. 관련 연구

2.1 이벤트 전개 모델

Sensor-Effector 모델, Rule-Based 모델, Predefined 모델 등과 같이 인공지능, 가상현실, 컴퓨터 애니메이션 등의 분야에서 다양한 상황 및 이벤트를 전개하기 위한 많은 모델들이 연구되어져 왔다. 이러한 모델들은 그전의 Path 모델에 비해 비교적 자연스럽게 다양한 상황을 제공할 수 있었으나 한정된 형태의 자율성과 방대한 탐색 공간의 문제를 가지고 있었다[3].

이러한 문제점을 해결하기 위해 Relation-Based Model이 제시되었다. Relation-Based Model은 객체들 사이의 관계들과 관계 내에 객체 사이의 행동(action)들을 미리 설정해 놓는다. 가상 환경에서 그 객체들이 출현하면 수많은 관계들 중에서 그 객체들과 관련이 있는 관계가 선택되어진다. 그 후 관계의 triggering condition이 만족되어지면 관계 내에서 정의되어진 행동이 실행되어진다. 이 모델은 한정된 도메인에서 다양한 상황을 처리하는 데는 적합하지만, 자연현상과 같이 많은 객체들이 등장하고 도메인이 넓은 분야에는 적용하기가 어렵다. 그 이유로 첫째는 수많은 객체들 사이의 관계를 미리 정하는 것이 어렵다는 것이다. 만약 관계들을 미리 정한다고 하더라도 그 자체가 다양한 상황을 제한하는 것과 같다. 둘째는 객체의 행동이 상황에 의해 설정되어지는 것이 아니라 관계의 설정시 설계자에 의해 임의로 정해진다는 점이다. 이것 또한 객체의 행위를 제한함으로써 다양한 상황의 전개를 어렵게 한다.

넓은 도메인에서 다양한 이벤트 전개 문제를 해결하기 위해 인과관계에 기반한 이벤트 전개 모델이 제시되어졌다. 이 모델은 인과관계를 제어 메커니즘으로 하는데, 상황 속에서 발생한 이벤트는 전개과정을 거치면서 상황에 영향을 주고, 그 이벤트의 효과는 또다시 다른 이벤트를 발생시킨다[5]. 즉, 이벤트의 효과와 다른 이벤트들 사이는 인과관계로 연결되어진다. 이 모델에서 이벤트의 전개는 상황 의존적이므로 자연 현상과 같은 규칙 기반의 다양한 현상 시뮬레이션에 적합하다[6]. 그러나, 이 모델은 객체가 고유의 성질 및 특징을 유지하고 있을 때에만 적용 가능하다. 이벤트에 참여하고 있는 객체가 정성적 변화에 의해 그 성질이 변했을 때, 발생하는 예측하기 어려운 이벤트의 처리에 대해서는 다루고 있지 않다. 자연 현상에서는 이러한 변화가 많이 존재하고 또한 이러한 변화로 인해 더욱더 예측하기 어려운 상황이 만들어진다. 그러므로 객체의 정성적 변화에 의한 이벤트의 전개 방법이 요구되어진다.

2.2 가상현실 및 게임에서의 가상 환경 표현

많은 게임에서도 가상 환경이 등장하고 그들 중 대부분은 우리가 실제로 보는 환경과 최대한 비슷한 모습과 느낌의 환경을 모사한다. 지금까지 게임이나 가상현실 분야에서

가상 환경을 설계하는데 있어서 주요 관심사는 시각적으로 얼마나 현실감이 있으면서 이를 실시간적으로 표현하는데 필요한 비용을 절감 할 수 있는가의 여부였다[7]. 그러나 그곳에서 발생하는 자연 현상들은 단순하거나 고정된 경로를 따라 임의로 발생한다. 그리고 시각적인 사실성을 높이기 위해 그래픽적인 기법에 치중하고 있다. 예를 들어서 빛에 의한 그림자의 발생은 삼차원적인 깊이를 느끼게 하는 요인으로서 중요시되고 있고, 여기에 초점을 두어 복잡한 모양의 객체에 대해서도 그림자를 생성하는 레이 캐스팅이나 래디오시티와 같은 알고리즘들이 이용되어져 왔다[8]. 그러나 그림자의 영향에 의한 다른 이벤트의 발생과 같은 논리적 사실성은 간과되어져 왔다. 이러한 논리적 사실성을 고려한 본 연구의 가상 환경 설계는 그래픽적인 기법과는 다른 관점에서 지금까지의 단순하고 고정된 이벤트의 흐름을 배제하고 다양하고 예측할 수 없는 상황을 전개하게 해준다.

2.3 기상 현상 시뮬레이션

기상 현상 시뮬레이션은 기상학 분야에서 많이 연구되어져 왔다. 기상 현상 시뮬레이션은 기상 현상의 과학적 분석을 통한 데이터와 지역의 특성을 나타내는 요소들을 충분히 적용하여 그 지역의 기후를 예측하는 것을 목적으로 한다. 그러므로 기상 현상 시뮬레이션은 정확한 예측을 위해 세밀한 모델링과 정확한 많은 데이터가 요구된다. 지금까지 수년간 GCM(General Circulation Model)에 대한 연구가 많았는데, GCM은 지구 전체에 걸쳐 있는 대기와 그것의 현상들을 수치적으로 시뮬레이션 하는 모델이다. 이를 위해 태양 복사, 광화학, 열전달, 물의 증발, 운동량 등에 관한 방정식을 활용한다. 최근에는 GCM의 중요한 부분으로 지표면과 식물, 공기 사이의 상호작용(surface-plant-air interactions)에 관한 수치적 시뮬레이션이 많이 연구되어지고 있다[9]. 이는 GCM에서 다루어진 넓은 영역에 대한 기상 현상 시뮬레이션과 달리 좁은 영역에 대한 지역적인 기후에 대해서 다루고 있다. 대표적인 모델로서 ENVI-net을 들 수 있으며, 도시지역 환경 내에서 시뮬레이션을 목적으로 만들어진 삼차원 비유체 정역학 모델로서 빌딩, 지표면, 식물들 사이의 상호작용을 분석하고 예측한다.

기존의 모델들은 현상에 관련된 모든 요인들(relevant factors)을 매개로 복잡한 계산을 수행 해 왔다. 하지만 본 논문에서 다루는 자연 현상 시뮬레이션은 위의 모델들과 같은 정확한 예측을 위한 것이 아니라, 가상현실에 있는 에이전트에게 현실과 비슷한 다양한 현상을 제공하는 것을 목적으로 한다. 이를 위해 환경 변수의 통계적 특성을 이용하여 환경을 Region으로 나누고 각 Region별로 현상과 관련된 주요 환경 변수를 계산한다. 이를 이용하여 환경 내의 각 요소들과 논리적인 연관성을 가진 다양한 현상들의 발생, 전개, 변화, 중단 등의 상황을 제공할 수 있다.

3. 객체의 구조 및 가상 환경

3.1 Physical Entity Modelling

가상 환경은 수많은 객체들로 구성되어진다. 그 객체들은 자신만의 성질을 가지고 있고, 또한 다른 객체들과 상호 작용한다. 본 논문에서는 관련된 ontology에서 제공하는 지식을 바탕으로 객체를 설계한다.

3.1.1 Ontology의 상위 레벨

Ontology는 주어진 응용 도메인에 따라 “세상을 보는 관점(World View)”을 달리한다[10]. 그러므로 수많은 Ontology들의 최상위층(root level)은 “thing” 또는 “entity”로 비슷하게 출발하나 다음 계층에서는 주어진 도메인에 따라 다양하게 확장해 나간다[11]. 본 논문에서는 최상위층을 물리적 개체(Physical Entity)로 하고, Physical Entity는 Physical Object, Physical Concept, Material을 포함한다.[12] Material은 세상을 구성하는 가장 기본적인 물질(substances)이다. 그것들은 절대로 사라지지 않으며 단지 모양이나 상태측면에서 변할 뿐이다. 그리고 Material은 자체로서 세상에 존재하거나, 또는 Physical Object를 구성하는 재료(ingredient)로 이용되어진다. Physical Object는 고체의 특성을 가지고 있고, 하나의 Material로 구성되어진 Basic Object와 그 Basic Object들을 구성 성분(components)으로 가지는 Composite Object로 나눌 수 있다. Physical Object는 부서지거나 또는 어떤 요인으로 인해 자신의 고유한 특성을 잃었을 경우 다시 Material로 돌아간다. Physical Concept은 에너지, 빛, 소리와 같이 물리적 성질을 가지고 있지만 공간의 배타적 점유를 하지 않는 것들을 일컫는다. 이들은 Physical Object, 또는 Material에 의해 발생되고, 자연 현상을 일으키는 근본적인 원인내지 매개가 된다.

3.1.2 Physical Object의 Attributes

Physical Object의 속성 (Attributes)은 모양, 질량, 부피, 공간적 위치와 같이 Physical Object 전체를 서술하는 Descriptive Attribute와 비열, 열 흡수율, 빛 반사율과 같이 Physical Object의 Ingredient(Material)가 가지는 Inherent Attribute로 나누어진다. Descriptive Attribute는 이벤트나 다른 Physical Object의 Activity의 영향에 의해 변화 할 수 있지만, Inherent Attribute는 Physical Object가 정성적 변화를 하지 않는다면, 바뀌지 않는 고유의 성질이다.

3.1.3 Physical Entity의 Activities

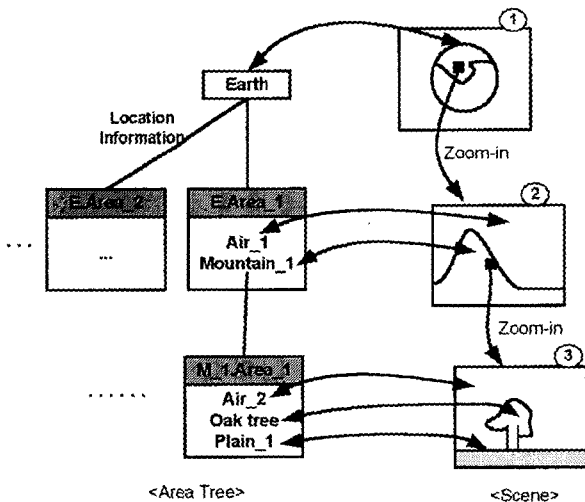
Physical Entity의 Activity는 그것의 상태를 변화시키거나, 또는 주위 환경 요소에 영향을 주는 행위들을 일컫는다. 자연 현상은 기본적으로 이러한 Activity들이 활성화되고 난 후, 실행(Execution)과정이나 또는 그 결과(Effects)에 의해 발생되어진다. 자연 현상과 관련된 Physical Entity, 즉 Physical Object 또는 Material은 외부적인 힘이나

열과 같은 Physical Concept이 작용하지 않으면 아무런 변화가 없다. 자연 현상은 이러한 외부적 요인으로 인해 Physical Object 및 Material의 Activity가 활성화되어지고, 또한 이러한 Activity들과 인과관계로 연결된 다른 Activity들이 재귀적으로(recursively) 연속적으로 활성화되어짐으로써 전개되어진다. 자연 현상에서 Activity는 철저하게 규칙들에 의해 활성화되어지고 전개되어진다. 규칙의 Antecedent 즉 condition clause는 환경변수나 Physical Entity나 그들 사이의 관계에 관한 Attributes로 표현되고, Consequent는 규칙의 효과부분으로서 Physical Entity의 Activity 또는 Event로 표현된다[13].

3.2 Physical Object의 Instance와 Area tree

Ontology를 바탕으로 구성된 class에 속하는 개체가 실제적으로 가상환경에 존재하기 위해서는 instantiation의 과정을 거쳐야한다. Instantiation은 두 단계의 과정을 거친다. 첫 단계는 기본적으로 class를 표현하는 attribute들의 값을 지정함으로써 이루어진다. 즉, 지금까지 OOP(Object-Oriented Programming)에서 다루어진 방법을 말한다. 두 번째 단계는 첫 단계에서의 Instance를 여러 구역(Area)으로 나누는 것으로서, 각 구역은 하나의 가상환경이 된다. 이는 그 Physical Object가 다른 Physical Entity들에게 공간을 제공한다라는 개념을 바탕으로 접근하는 방법이다. 즉 구역은 Physical Object가 제공하는 하나의 공간을 의미하고 상황에서 하나의 장면(scene)을 나타낸다.

Physical Object의 구역은 설계자의 의도에 따라 여러 계층들을 가지는 tree로서 구성되어야 할 수 있다. 이를 Area tree라고 부르고 하나의 Area_node는 하나의 장면에 대응되면서 장면에 구성되어져 있는 가상환경의 정보를 가지는 frame이다. Area_node에 속해 있는 Instance들은 가상환경을 구성하는 요소들이다.



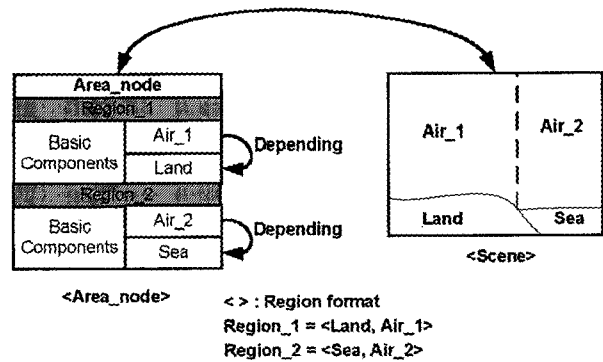
(그림 1) Area tree와 장면사이의 관계

Area_node는 기본요소들(Basic Components)과 추가요소들(Additional Components)로 구성된다. 기본요소는 Area_node가 종속되어진 상위 계층의 instance의 전체 및 부분을 말한다. 추가요소는 임의로 추가된 다른 Physical Object 또는 Material의 Instance를 말한다. (그림 1)에서 'M_1.Area_1' Area의 기본요소는 'Plain_1'과 'Air_2'이다. 'Plain_1'은 'Mountain_1'의 요소 중에서 하나이고, 'Air_2'는 'Earth'의 요소 중에서 하나이다. 'Air_2'는 실제적으로는 'Earth'의 요소의 하나이지만, 'M_1.Area_1'이 간접적으로 'Earth'에 종속되어져 있으므로 'Air_2'는 'M_1.Area_1'의 기본 요소가 될 수 있다. 그리고 'Oak tree'는 추가 요소이다. 'Oak tree'는 'Mountain_1'의 요소가 아닌 독립적인 하나의 Instance이지만, 'M_1.Area_1'을 구성하고 있다.

4. 가상 환경에서 자연 현상의 모델링

4.1 Area_node에서 Region

기존의 자연현상 모델링에 관한 연구에서는 공간을 분할한 공간 객체를 이용하여, 모든 공간 객체에 있는 기온, 기압, 공기 밀도, 포화 수증기량과 같은 환경 변수들을 계산하였다.[6] 이 방법은 계산량이 많을 뿐만 아니라 필요하지 않은 계산까지 포함하고 있다. 본 논문에서는 이러한 계산량을 줄이기 위해 환경 변수의 통계적 특성을 이용한다. 예를 들어 기온의 수평적인 분포는 지역(Region)의 특성을 이용한다. Region이라는 것은 가상 환경에서 환경의 특성이 비슷한 지역을 말한다. 비슷한 특성을 가진 지역에서는 지표면의 기온이 수평적으로 차이가 없다. 이러한 특성을 이용하여 한 지역내에서는 지표면의 기온이 같다고 본다. 하나의 Area_node를 구성할 때도 이러한 특성이 반영되어야 한다. (그림 2)는 Area_node에서 지리적 특성을 이용하여 지역을 형성하는 것을 보여준다.



(그림 2) Area_node에서 지역 형성

'Air_1'은 'Land'에 의해 기온이 결정되어지고, 'Air_2'는 'Sea'에 의해 결정되어진다. 즉 대기의 특성이 해당 지표면의 특성에 의해 결정되어지는 것이다. 이런 점을 이용하여

Area_node에서 두 개의 지역, 즉 'Region_1'과 'Region_2'를 만들 수 있다. 하나의 Area_node는 크게 Region들로 나누어지고, 그 Region들은 각각 Basic Components와 Additional Components로 구성되어진다. (그림 4.2)는 Additional Components가 없는 경우이다. 만약 하나의 Area_node가 여러 개의 지역으로 구성되어지면 지역별로 환경 변수의 계산을 하면 된다.

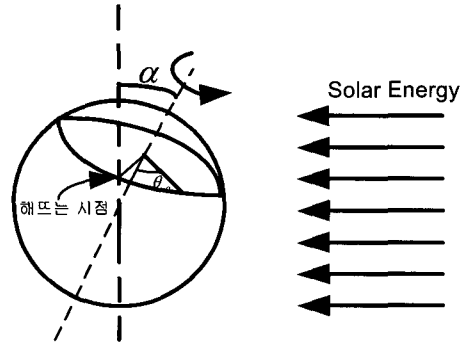
4.2 태양 복사(Solar Radiation Event)

지구에서의 대부분의 자연현상은 직간접적으로 태양에서 방출된 태양 에너지에 의하여 발생한다. 그러므로 태양 복사 과정은 자연현상에서 중요하면서도 기본적인 것이다. 여기서는 태양 복사 과정을 두 가지 관점에서 모델링한다. 즉 태양빛이 태양에서 지구까지 오는 거시적 관점에서의 과정과 어떤 지역의 대기에서부터 지표면까지 오는 미시적 관점에서의 과정이다. 태양복사 현상의 과정은 (그림 3)과 같다. 'Solar radiation' 이벤트에서는 위도와 경도에 의한 에너지량의 결정, 그리고 그 에너지에 의한 지표면 온도의 상승, 빛의 입사각에 의한 밝기 조절, 빛의 이동 중 객체와의 충돌과 그 충돌에 의한 그림자 영역의 생성과 소멸에 초점을 맞춘다.

거시적 관점(macroscopic view)에서 태양에서 지구까지 도달하는 에너지의 양은 일정하다고 가정한다면, 지구에 있는 어떤 지역에 도달하는 에너지의 양(E_0)은 위도와 경도에 의해 결정되어진다. 지표면에서의 도달 에너지는 다음과 같이 수식화 할 수 있고 관련된 factor들은 (그림 4)에서 표현되어져 있다. (상세한 수식들 및 설명은 [14] 참조.)

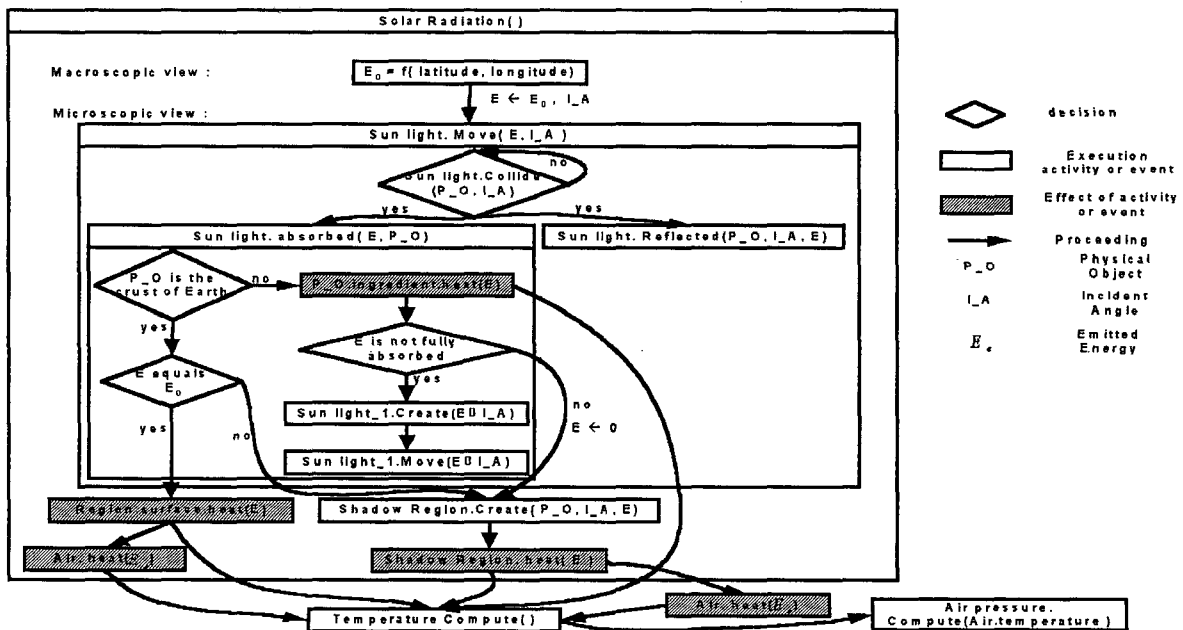
$$E(t, s) = \cos(\theta_i - \alpha) \frac{\sin \theta_a + \sin(\alpha \theta_i / 40)}{1 + \sin(\alpha \theta_i / 40)} E_0 \quad (1)$$

α : 지구의 기울어진 축의 각도 ($-23.5 \leq \alpha \leq 23.5$)
 E_0 : 태양에서 지구까지 도달하는 열에너지(단위 : 칼로리)
 θ_a : 축을 중심으로 했을 때 회전각(그림 4), θ_i : 위도



(그림 4) 태양 복사

식 (1)에서 ' $\cos(\theta_i - \alpha)$ '는 지구의 공전과 관계가 있는 부분이다. 즉, α 값이 변화함에 따라서 그 위도에 도달하는 최고 에너지의 비율(0~1)을 계산하는 부분이다. θ_a 는 지구의 자전에 따라 변화한다. (그림 4.4)에서처럼 θ_a 는 점선의 기울어진 축을 중심으로 계산되어진 각이다. 예를 들어서 위도가 0°인 지점은 해가 뜨는 시점이 θ_a 가 0°일 때이다. 그러나 위도가 0°보다 높고, 여름으로 다가가는 시점에서 해가 뜨는 시점이 더 빨라진다. 해가 뜰 때와 해가 질 때의 θ_a 의 값은 다음과 같이 볼 수 있다.



(그림 3) 태양 복사 이벤트의 과정

$$\text{해가 뜰 때} : \theta_a = -\alpha \theta_i / 40 \quad (2)$$

$$\text{해가 질 때} : \theta_a = 180^\circ + \alpha \theta_i / 40 \quad (3)$$

(그림 3)에서 'Sunlight.move'는 태양 빛의 이동을 나타내는데, 만약 이동 중 객체와 충돌을 하게 되는 경우 그 객체가 지표면이면 열에너지는 전달되어진다. 여기서 충돌한 객체가 지표면인지를 판단할 수 있는 방법은 그 객체가 Area_node의 기본요소에 속하는지를 통해서 알 수 있다.

태양복사에 의한 에너지의 변화율 E_p' 과 그에 따른 지표면의 온도 T_p 는 아래와 같이 수식화할 수 있다. 식 (3)에서 보는 바와 같이 T_p 는 일정한 ΔT 동안에 축적된 값이다. (상세한 설명은 [14] 참조.)

$$E_p'(t, s) = E(t, s) - k(T_p(t, s) - T_a(t, s)) \quad (2)$$

(단, $T_p \geq T_a$)

$$T_p(t, s) = \int_i^{i+\Delta T} \frac{E_p'(t, s)}{k_p} dt + T_{p0} \quad (3)$$

E : 지표면에 입사하는 에너지

k : 공기의 비열 k_p : 지표면을 구성하는 재료의 비열

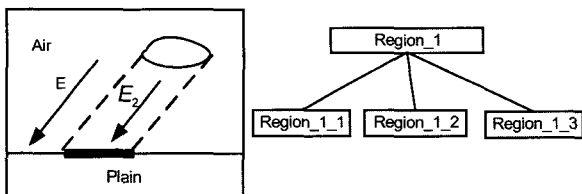
E_{p0} : 지표면의 초기 에너지 값

E_a : 공기의 초기 에너지 값

T_p : 지표면의 온도, T_a : 공기의 온도

만약 구름과 같은 객체가 대기에 존재한다면 그 객체에 의해 반사내지 흡수되어지고, 그로 인해 지표면에 도달하는 에너지의 양이 감소되어짐으로써 그림자 영역(shadow region)이 발생한다. 이 과정은 (그림 3)에서 보는 바와 같이 'Sunlight.collide()'에 의해 전개되어진다.

(그림 5)에서 'Region_1_2'가 그림자 영역이다. 유의할 점은 'Region_1'은 삭체가 되지 않고 계속해서 존재해야 한다는 것이다. 지금 현재 그림자 영역이 사라지면 분할된 영역들은 사라지고 다시 'Region_1'이 모든 환경변수들을 관리해야 하기 때문이다. 다시 말해 분할된 영역들은 그림자로 인해 발생한 가상영역(Virtual Region)들이다. 가상영역들이 형성된 경우에 태양 복사 이벤트는 가상영역에서 진행되어진다.



(그림 5) 그림자 영역 발생시 area_node의 region 분할

처음에 가상영역들이 형성되면 각각의 환경변수들 값은 원래 영역의 환경변수 값들로 초기화되어진다. 예를 들어서

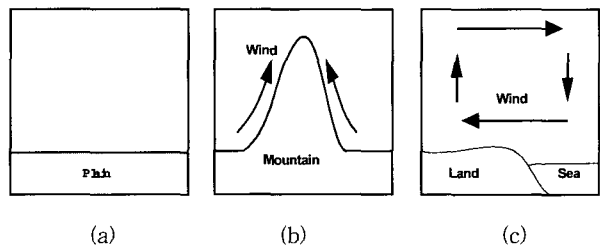
구현에서 'Region_1_1'의 Plain의 온도는 다음과 같이 초기화된다.

$$\text{Region_1.Region_1_1.Basic Component.Plain.ingredient. temperature} = \text{Region_1.Basic Component.Plain.ingredient.temperature};$$

그림자 영역의 소멸 조건은 두 가지가 있다. 첫째는 빛과 충돌하는 대상이 없어지는 경우이고, 둘째는 빛이 없어지는 경우이다. 두 가지 조건 중 하나를 만족하면 그림자는 없어진다. 이때 'Region_1_1', 'Region_1_2'와 'Region_1_3'도 삭제되는데, 삭체가 되기 전 그들이 가지고 있는 환경변수들의 평균값을 'Region_1'의 각 환경변수에 업데이트 해주어야 한다.

4.3 바람 현상(Wind Event)

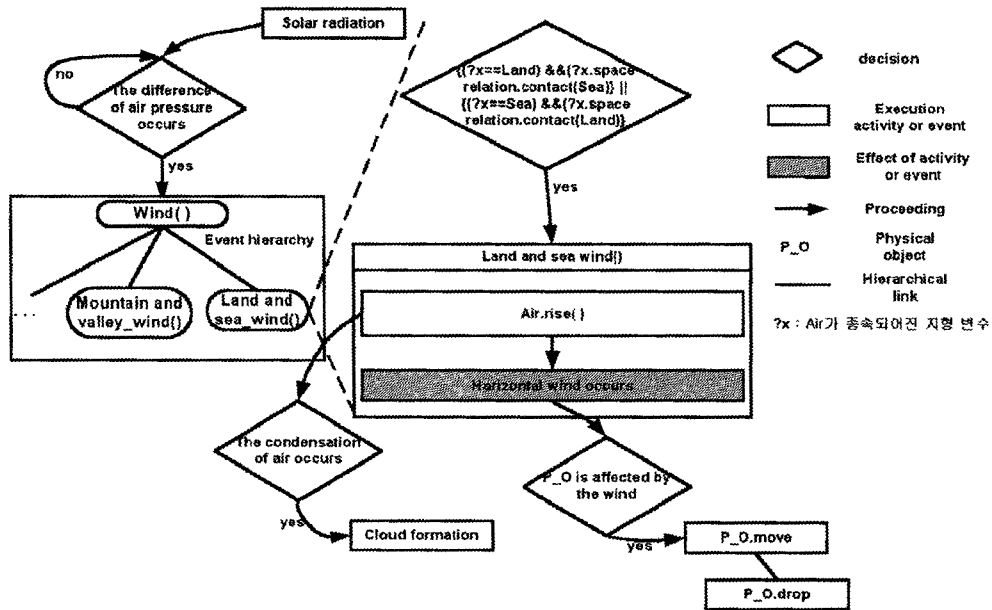
바람은 기압차에 의해 발생한다. 앞에서 다루었던 것처럼 하나의 영역에서는 수평적으로 기온의 차이가 없다고 가정하였다. 그러므로 서로 다른 영역이 접하게 되면 기온의 차이가 발생하게 되고, 이것으로 압력의 차이가 발생하고 결과적으로 바람이 발생하게 된다. 또한 그러한 바람의 진행은 공간적 상황에 따라 다르다. (그림 6)(a) 그림은 하나의 영역만 존재하는 경우이다. 이 경우에는 지표면의 기온의 차이가 발생하지 않으므로 바람이 발생하지 않는다. (그림 6)(b)와 (그림 6)(c)의 경우에는 두 개 이상의 영역이 존재함으로써 바람 발생의 전제조건(precondition)을 만족하게 되고, 발생한 바람의 진행은 공간적 상황에 따라 다르게 진행되어진다.



(그림 6) 공간 요소에 의한 바람의 진행

공간요소들(spatial factors)에 의한 바람의 진행은 Wind Event Hierarchy를 사용하여 표현한다. Wind Event Hierarchy의 예제와 바람의 진행 과정은 (그림 7)과 같다.

예를 들어 (그림 6)(c)과 같은 가상공간에 대한 area_node의 프레임 구조는 (그림 8)(a)와 같다. (그림 8)(a)의 상황에서 태양복사 이벤트에 의해 Region_1과 Region_2의 지표면의 기온이 달라지면 기압도 달라진다. 또한 area_node에 두 개의 영역만 있으므로 기압이 높은 지역과 기압이 낮은 지역을 포함한 바람의 영향권에 있는 영역을 결정하기가 쉽다. 기압이 높은 지역을 s_H 라 하고 기압이 낮은 지역을 s_L 이



(그림 7) Wind Event Hierarchy 구조와 이벤트 진행과정의 예

라고 부르겠다. 그러나 여러 영역이 존재하는 경우에는 바람의 영향권에 있는 영역이 s_H 와 s_L 두 영역만 있을 수도 있고 여러 영역이 있을 수 있다. 예를 들어, (그림 8)(b)와 같이 태양복사 과정에서 구름에 의해 원래의 영역이 4개의 기상영역들로 나누어지는 경우의 예를 보자. (b)에서 'Region_1_2'는 'Region_1'에 속하면서 그림자 영역에 해당한다. 마찬가지로 'Region_2_1'은 'Region_2'에 속하는 동시에 그림자 영역에 해당한다. 이런 상황에서 바람이 (b)의 그림처럼 분다는 것은 s_H 는 'Region_2_1'이 s_L 은 'Region_1_1'이고 'Region_1_2'는 s_H 와 s_L 어디에도 속하지는 않지만 바람의 영향권에 속한다는 것을 의미한다.

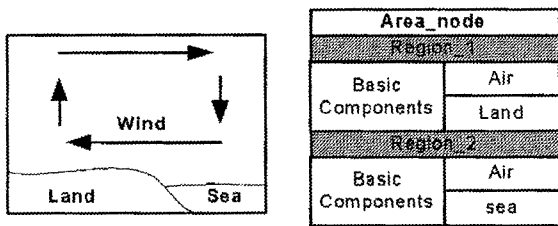
구체적으로 바람이 부는 영역에 어떤 물리적 객체가 존재한다면 그 객체에 미치는 바람에 의한 힘은 기본적으로 다음과 같이 표현된다.

$$\vec{F} = A|P(s_H, t_a) - P(s_L, t_a)| \overrightarrow{s_H s_L} \quad (4)$$

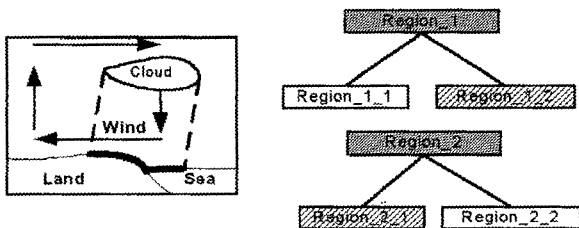
단, A 는 바람의 영향을 받는 물체의 단면적, $P(t, s)$ 는 공간 s 에서 시간 t 에서의 기압, s_H 의는 기압이 높은 지역, 그리고 s_L 은 기압이 낮은 지역을 말한다. 위의 기본적 식을 바탕으로 여러 가지 바람에 관련된 현상들을 유도할 수 있다. 예를 들어 바람에 의한 물리적 객체의 수평 이동거리 s 는 아래와 같이 구해질 수 있다.

$$s = \frac{A|P(s_H, t_a) - P(s_L, t_a)|}{2m} (t_2^2 - t_1^2) \quad (5)$$

식 (5)에서 m 은 물리적 객체의 질량, t_1 과 t_2 는 힘을 받은 객체가 움직인 시간 구간을 나타내고, t_a 는 힘이 가해진 시점이다. 그리고 이동 방향은 식 (4)에서 $\overrightarrow{s_H s_L}$ 에 의하여 구해진다[14].



(a)



(b)

(그림 8) 바람이 발생하는 공간 상황의 예와 표현

4.4 Physical Object의 정성적 변화 모델링

Physical Entity의 정성적 변화는 어떤 외부적 내부적 요인에 의해 Physical Entity가 자신의 성질을 잃고 다른 성질을 가지는 Physical Entity로 변질(transmutation) 되는 것을 말한다. 만약 어떤 Physical Entity가 Event에 관련되거나, 자신의 Activity로 인해 주위에 영향을 주는 상황에서 이 Physical Entity가 정성적 변화를 일으킨다면 예측하지 못한 상황이 전개되어질 수 있다. 예측하지 못한 상황이

라는 것은 진행 중이던 Event가 중단되거나, 변질 전에 그 Physical Entity의 Activity에 의해 영향을 받던 다른 Physical Entity가 변질 후에 영향을 받지 못함으로 인해 어떤 반응을 하는 경우를 말한다. 또한 추가적인 조건에 따라 주위에 영향을 줄 수도 있다.

4.4.1 Material의 변화

Material의 상태 변화는 어떤 Material이 다른 상태로 변화하는 것을 말하고 그 변화가 제한적이다. 예를 들어서 'water'의 상태 변화는 온도에 의해 결정되어진다. 온도 조건만 맞으면 'water'는 'ice'나 'vapor'로 변화한다. Material 상태 변화가 제한적이라는 특성을 이용하여 Material의 class에서 이들 사이의 관계를 미리 설정하는 것이 가능하다. 상태 변화보다 광범위한 물질 변화로서 Material이 상황 변화에 반응하여 다른 Material로 변화하거나 Material들이 상호 반응하여 새로운 물질이 생성되는 경우도 포함한다.

4.4.2 Basic Object에서 Ingredient 상태 변화

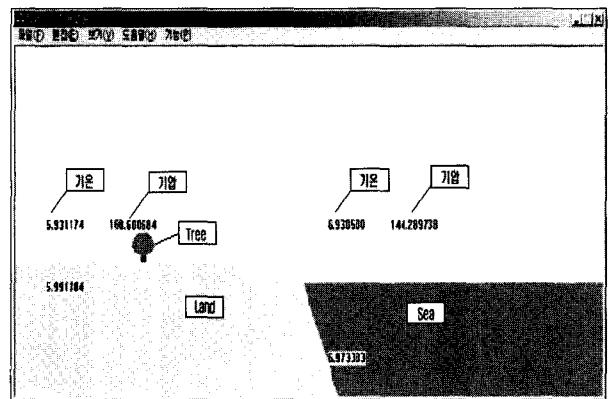
앞에서 언급한 바대로 Material은 그 자체로서 존재하거나 Physical Object의 원료(ingredient)가 될 수 있다. 여기서는 Basic Object의 ingredient인 Material이 변화를 했을 때 그 결과로서 Basic Object의 변화가 주요 관심사가 될 수 있다. 첫째 Basic Object의 Inherent attributes의 값이 변화한다. 즉, Ingredient인 Material이 변화했으므로 새로운 Material의 Attributes가 Inherent attributes가 된다. 둘째는 그 Basic Object의 Activity가 모두 종결, 즉 Basic Object가 Activity의 능력을 잃게 된다. 셋째는 Basic Object가 다른 타입으로 바뀐다.

(그림 9)를 보면, 처음에는 'Material_1'이 'Basic Object_1'의 Ingredient이었다. 그러나 Ingredient의 상태 변화에 의해 'Material_1'이 'Material_2'로 변화하고, 결과적으로 'Basic Object_1'의 Ingredient가 바뀌었다. 이에 따라 'Basic Object_1'의 Activity가 모두 종결되어진다. 이로 인해 기존에 이 Activity의 영향을 받던 Physical Entity들은 그 영향이 사라짐으로 인해 다른 상황을 만들어 낸다. 이 부분은

구현 및 테스트에서 다시 한번 다루도록 하겠다.

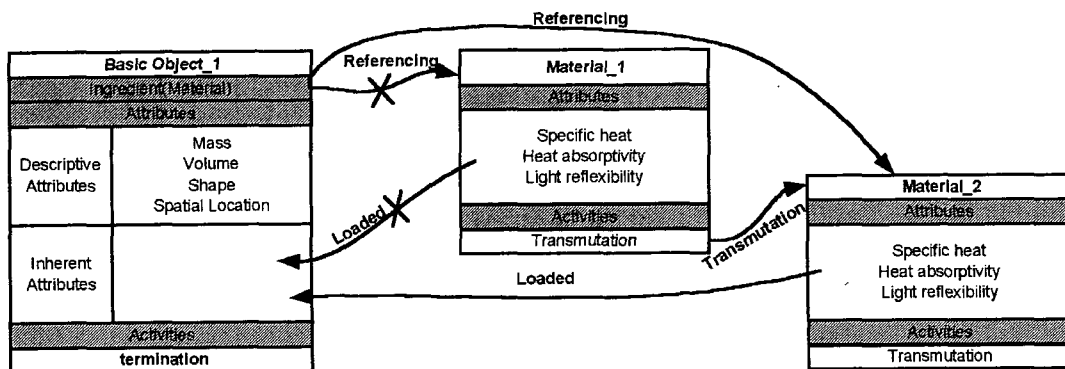
5. 구현 및 고찰

지금까지 태양복사와 그에 관련된 대표적 자연 현상들을 제안된 modelling 방법들을 기초로 구현한다. 이를 통하여 다양한 상황들이 논리적으로 전개될 수 있음을 보이고자 한다. 구현은 Windows 환경에서 Visual C++을 사용하였다. 가상 환경의 초기화면은 (그림 10)과 같다.

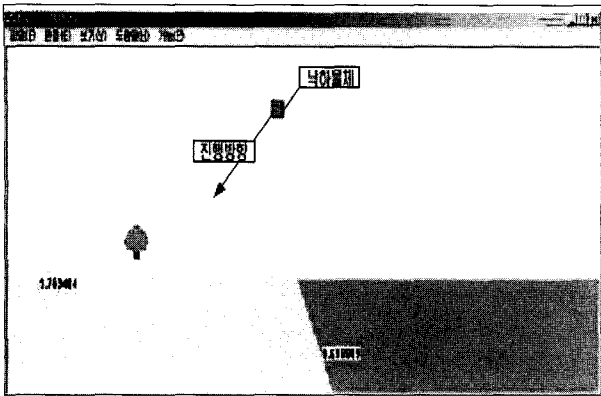


(그림 10) 시뮬레이션의 초기 화면

(그림 10)에서 보이는 환경은 위도30°, 경도45°인 지역이다. 화면에 보이는 수치들은 태양복사 이벤트에 의해 변하고 있는 기온과 기압의 값들의 순간치들이다. 그리고 이 환경의 정보를 관리하는 Area_node는 "Land"와 "Sea" 두 개의 영역으로 구성되어 있다. 그리고 "Land" 영역에는 추가 요소로써 나무(tree)가 하나 추가되어 있다. Land와 Sea는 지구의 요소들로서 area_node에서는 기본요소로 사용되어지고 있다. 그리고 현재 눈에는 보이지 않지만 기화(Evaporation) 이벤트가 진행 중이고, 또한 두 개의 다른 성질의 영역이 접해 있으므로 Wind 이벤트도 진행 중이다. 이것을 테스트하기 위해서 (그림 11)에서 보이는 것처럼 하나의 물체를 이 환경에 임의로 낙하 해본다.

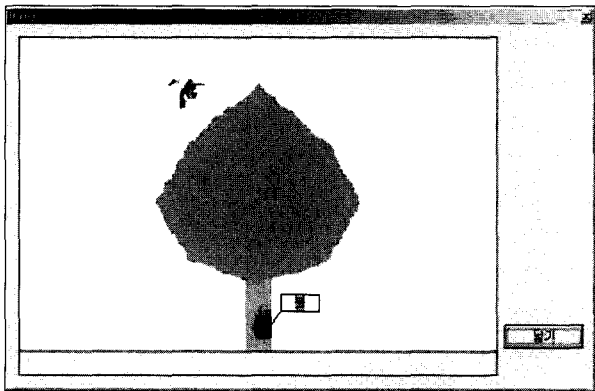


(그림 9) Ingredient의 상태 변화에 의한 Basic Object의 변화



(그림 11) 낙하 테스트

낙하 물체는 임의로 질량을 설정할 수 있는데 여기서는 바람의 영향을 쉽게 받기 하기 위하여 가벼운 질량을 선택 하였다. (그림 11)에서 바람이 존재하지 않는다면 현재의 낙하물체는 수직 하강 할 것이다. 그러나 현재 바람이 Sea Region에서 Land Region으로 불고 있기 때문에 물체는 표시된 방향으로 진행하고 있다. 물론 질량이 크면 클수록 바람의 영향을 적게 받기 때문에 수평으로 움직이는 폭이 작아 질 것이다.

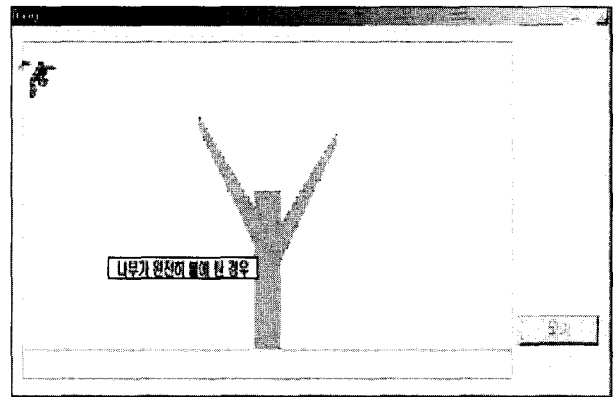


(그림 12) 나무의 화재에 의한 새의 이동

(그림 12)는 (그림 11)의 tree를 줌인 했을 때 나타나는 새로운 가상 환경이다. 즉, 나무를 최상위 계층으로 하는 구역이 미리 설계되어져 있다는 의미이다. 이 환경을 관리하는 Area_node의 기본 요소는 Land이고 나무와 새(bird)는 추가요소들이다. 현재 나무의 Activity중에서 과일과 잎을 생산하는 Activity가 새에게 식량으로 제공되어지고 있기 때문에 새라는 Agent는 나무라는 환경을 벗어나지 않고 그 환경에 반응하며 존재하고 있다. 이 시뮬레이션의 목적은 화재에 의해서 나무가 정성적 변화를 일으키게 되고 이로 인해 나무와 새의 관계가 깨어짐으로써 새가 나무를 떠나는 예상치 못하는 상황이 발생한다는 것을 보여주는 것이다. 만약 나무라는 Physical Entity가 어떤 외부적 요인에 의해 정성적 변화를 수행하게 되면 Physical Entity는 Attri-

bute와 Activity를 잃어버리게 되고 그로인해 Agent와의 관계도 깨어질 것이다. (그림 12)의 상황은 나무에 살고 있는 새가 화재에 의한 결과로 자신의 거주 지역을 벗어나는 장면이다. 현재 화재에 의해 나무(wood)가 숯(charcoal)으로 정성적 변화를 하는 중이다. 이 새는 완전히 거주지를 버리지 않고, 위험이 사라질 때까지 나무의 주위를 맴돌 것이다.

(그림 13)은 화재에 의해 나무가 완전 정성적 변화를 한 경우를 보여준다. 즉, 나무가 숯으로 변화한 것이다. 이로 인해 나무가 전에 가지고 있던 Activity는 사라진다. 그 결과 새가 나무를 버리고 다른 곳으로 떠나는 예기치 못한 상황이 발생하게 된다.



(그림 13) 나무의 정성적 변화

6. 결론 및 향후 연구 방향

본 논문에서는 가상 환경에서 태양복사와 바람 이벤트를 논리적으로 시뮬레이션 하는 것을 목적으로 한다. 이들 이벤트 진행과 이벤트들 사이의 연결을 인과관계 및 규칙들에 기반하여 전개함으로써 논리적 사실성을 높이고 또한 상황의 전개가 다양하고 예측할 수 없도록 하는 것이 근본적인 목적이다. 이를 위해 우선적으로 Ontology적 지식을 바탕으로 하여 class를 구축하고 이것을 두 단계의 instantiation 과정을 거치게 한다. 첫 번째 단계에서의 instantiation은 일반적인 과정이고 두 번째 단계에서는 이 instance들을 Area_node 즉 가상 환경에 출현시키는 과정이다.

여기서 제안한 방법의 장점은 다음과 같다. 첫째, 규칙들에 기반하여 이벤트들을 구성함으로써 그 전개가 다양하고 예측하기 어렵다는 것이다. 또한 객체의 정성적 변화를 처리하는 방법을 제시함으로써 좀더 예측하기 어려운 상황을 전개할 수 있다. 둘째, Area tree는 설계자의 임의로 언제든지 확장이 가능하다. 필요에 따라 수평적, 수직적인 Area_node를 추가함으로써 보이는 장면을 늘일 수 있다는 것이다. 셋째는 Area_node의 재사용 및 확장이 가능하다는 것이다. 이미 만들어진 환경에 새로운 객체를 추가 할 수 있다. 단 본 논문에서 다루지 않은 객체들 사이의 공간관계가

추가된 경우에 가능하다.

위의 원리에 기초하여 구현된 가상환경에서는 태양 복사와 바람 현상을 보여주었다. 시뮬레이션된 이벤트들에 관한 여러 현상이 시각적으로 보이지는 않지만, 영향을 받을 수 있는 개체들을 투입해 봄으로써 그 존재를 확인 할 수 있다. 특히 wind 이벤트는 설계자도 예측할 수 없는 방향으로 다양한 전개를 보여주었다. 또한 객체들 사이에 관계를 맺고 있는 상황에서 하나의 객체가 정성적 변화를 하였을 때 발생하는 예측할 수 없는 상황도 시뮬레이션에 고려하였다.

본 논문에서는 정성적 변화 중에서 Entity의 존재성의 변화, Class Hierarchy에서 소속 class의 변화 등은 고려하지 않았다. 또한 완전 정성적 변화가 아닌 부분 정성적 변화에 대한 연구도 필요하다. 그리고 앞에서 언급한 것처럼 객체들 사이의 공간 관계를 쉽게 표현할 수 있다면 새로운 객체의 추가 및 삭제가 가능해지기 때문에 좀더 예측하기 어려운 상황을 전개할 수 있다.

참 고 문 헌

[1] Junji Nomura, "Virtual reality technology and its industrial applications," IEEE Control Engineering Practice, Vol.7, pp. 1381-1394, 1999.

[2] Pioch, N. J, et al, "A Virtual Environment for Learning to Pilot Remotely Operated Vehicles," IEEE Virtual Systems and MultiMedia, 1997, VSMM '97. Proceedings., International Conference, pp.218-226, Sept., 1997.

[3] 김노순, "가상세계 구축을 위한 환경모델링", 경북대학교 석사학위논문, 2000.

[4] Hanqiu Sun, "A relation-based model for animating adaptive behavior in dynamic environments," IEEE Transaction on Systems, Man and Cybernetics-Part A : Systems and Humans, Vol.27, No.2, pp.235-243, March, 1997.

[5] S. J. Ji, et al., "A Structured Causal Graph-Based Model for Event Development," CATE2001, Banff, Canada, pp. 236-240, 2001.

[6] 우영욱, "스크립트를 이용한 가상 세계에서의 자연현상처리 방법의 개발", 경북대학교 석사학위논문, 2001.

[7] Mark Deloura, et.al, "Game Programming Gems," Charles River Media, Inc., pp.550-557.

[8] Hyewon Seo, Thalmann, N. M, "LoD Management on Animating Face Models," IEEE Virtual Reality, 2000. Proceedings, pp.161-168, March, 2000.

[9] Bruse, M., Fleer, H., "Simulating surface-plant-air interactions inside urban environments with a three dimensional numerical model," Environmental Modelling and Software, Vol.13, pp.373-384, 1998.

[10] Nianbin Wang, Xiaofei Xu, "A Method to Build Ontology," High Performance Computing in the Asia-Pacific Region, 2000, Proceedings, The Fourth International Conference/Exhibition on, Vol.2, pp.672-673, May, 2000.

[11] Chandrasekaran, B., et al, "What Are Ontologies, and Why Do We Need Them?," IEEE Intelligent Systems, Vol.14, pp.20-26, Jan-Feb., 1999.

[12] Park, J., "Modeling physical entities," Kyungpook Nat'l univ., AIMM Lab., 2002.

[13] Park, J., "Modelling productive rules in cyber world," Kyungpook Nat'l univ., AIMM Lab., 2002.

[14] 조진영, "가상환경에서 Solar radiation과 Wind 이벤트의 시뮬레이션", 경북대학교 석사학위논문, 2002.



조진영

e-mail : jycho@dreamsoft21.com
 2000년 경북대학교 졸업(공학사)
 2003년 경북대학교 졸업(공학석사)
 현재 드림소프트 주식회사 주임 연구원
 관심분야 : 인공지능, 가상현실



박종희

e-mail : jhpark@ee.knu.ac.kr
 1979년 서울대학교 졸업(공학사)
 1981년 한국과학원 졸업(공학석사)
 1990년 Univ. of Florida 졸업(공학박사)
 현재 경북대학교 전자공학과 교수
 관심분야 : 멀티미디어 응용, Computer Aided Education, CAD/CAM, 지능형 정보 시스템, 분산 데이터 처리 시스템