

# 효율적인 게임 개발을 위한 사건 기반의 상황 계층 구조

박 정 용<sup>†</sup>

## 요 약

논문은 컴퓨터 게임 구현을 위한 상황 시뮬레이션 구조를 제안하고, 이를 바탕으로 게임 전개에 대한 분석을 가능하게 한다. 기존의 게임에 대한 연구는 물리엔진의 성능향상, 온라인 게임에서의 네트워크 부하 문제에서부터 물리현상을 자연스럽게 표현하는 문제 등에 대한 연구가 주를 이루고 있다. 본 논문에서는 일반적인 컴퓨터 게임구조에 대한 높은 수준의 상세화를 시뮬레이션 하기 위해서 상황 계층 구조를 제공한다. 논리적으로 전개되는 시뮬레이션 환경은 상황 계층 구조에 기반해서 사건과 상황을 정의한다. 사건은 사용자에 의한 명시적 사건과 시스템내의 반응에 의한 묵시적 사건으로 분류한다. 게임 발생에서 사건의 원천을 정의하고, 객체의 존재유무가 게임 진행의 주요한 요인임을 설명한다. 이러한 접근방법의 장점은 게임 시뮬레이션을 위한 개념적인 설계와 상황분석을 제공한다. 그리고 게임의 구현에 있어서 시뮬레이션 설계의 복잡성과 추상화 작업에 따른 복잡함을 감소시킨다. 특히, 사건정의를 통하여 게임 설계에 대한 분류적인 접근이 아닌 구조적인 접근이 가능하다. 제안하는 방법으로 여러 게임 중의 하나인 바둑돌 뺑기 게임의 예를 적용하여 구현 가능함을 보인다.

## Hierarchy Structure of Situation based on Event for Effective Game Development

Jung-Yong Park<sup>†</sup>

## ABSTRACT

This paper proposes a Situation Simulation Framework for implementing computer game and describes the possibility of analyzing game with unfolding situation. In last few years game systems have been evolving from the performance of physical engine, network traffic in the on line game to the representation of rendering physical phenomena. In computer game, a situation hierarchy structure which allows the designer for simulating high-level specifications of game structure. Logically simulated environment is created by defining situations and events based on hierarchy structure of the situation. We classify events into explicit event is occurred by user and implicit event is occurred by system. Our study defines the existence of objects is the most prevalent factor applied to any event in game world. The advantages of this approach are able to allow for providing the conceptual design for simulation game and analyzing the situation in the game world. And this method allows us to decrease the complexity of system design and abstraction modeling for the simulation game. Specially, the introduction of the definition of events allows us to approach game design in a structural manner rather than by their classification. The proposed method was implemented in the "Shooting BaDuk" among games.

**Key words:** Game(게임), Simulation(시뮬레이션), Multimedia(멀티미디어)

※ 교신저자(Corresponding Author) : 박정용, 주소 : 경북  
경산시 진량읍 내리리 15번지(712-714), 전화 : 053)850-  
4469, FAX : 053)850-4696, E-mail : jypark@daegu.ac.kr

접수일 : 2006년 12월 6일, 완료일 : 2007년 3월 6일

<sup>†</sup> 정희원, 대구대학교 전자공학부

## 1. 서 론

컴퓨터를 통해 이루어진 가상세계에서의 응용 분야는 컴퓨터게임, 게임을 이용한 교육용 시스템, 교육용시뮬레이션 분야 등에서 연구가 이루어지고 있다.[1-10] 이러한 분야에서의 실세계 상황 시뮬레이션의 구현은 많은 수학적 계산과 물리적 요소에 지능적인 행동을 부여하기 위해서 많은 작업량이 요구되어지고 있다.[11-14] 그리고, 게임 및 시뮬레이션 기반의 교육 시스템에서는 전체적인 시스템의 구조에 대한 분석 및 설계를 모듈식의 접근 방법이 주로 사용되고 있다. 그러나 실제적인 구현측면에서는 이러한 모듈식의 구조 분석 및 설계는 시공간적인 변화를 수반하는 게임 및 시뮬레이션의 특성상 개별 객체와 상황을 구체화 시키는데 부족하다. 왜냐하면 상황의 전개는 유동적이며, 각 상황에서 객체들의 행동은 능동적이며, 근본적으로 실세계를 반영해야 하는 특징을 가지기 때문이다. 즉, 실세계의 반영은 객체들의 복잡성과 그들 사이의 무수한 연관관계들의 다양성과 사건의 불 예측성이다. 또한 게임 진행의 근본적인 발생에 대한 원인 규명의 부족으로 게임 설계에 있어서 많은 어려움을 초래한다. 따라서 본 논문은 게임을 사건에 기반하여 분석하고 설계하는 방법으로 보다 효율적으로 구현 가능성을 보이고, 임의의 게임 제작시 분석 및 설계를 위한 개념적인 접근 방법을 제공한다. 이를 위해서 본 논문에서는 실세계 환경의 시뮬레이션을 위한 기본 구조는 집단화 관계로 표현될 수 있는 상황 계층구조로 정의하며, 상황 계층구조를 이루는 하위 단계의 사건을 세분화한다. 사용자에게 의해 발생하는 명시적 사건과 시스템 내에서 사용자와 상호작용에 의해서 인식된 다음 행동의 양상으로 나타나는 묵시적 사건으로 분류한다. 그리고 현실세계 환경의 시뮬레이션을 위해서 다음 사항들에 중점을 둔다. 첫째, 시뮬레이션 상황은 결과적으로 무수한 객체들의 속성 값의 변화로 이루어진다. 즉, 사건으로 구성된다. 따라서 개념적, 논리적 실세계 반영을 위해 사건을 정의하고 사건발생의 원인대상에 따라 세분화하여 관련성을 규명한다. 둘째, 사건, 객체 그리고 속성과의 관계를 정의한다.

마지막으로 실세계 상황 시뮬레이션의 전개에서 객체의 존재유무가 가장 중요한 요인임을 보인다. 제안하는 게임 상황 시뮬레이션 방법은 사건에 기초하

여 구현 가능성을 설명한다.

본 논문의 구성은 2장에서는 본 연구와 관련된 연구들을 살펴본다. 3장에서는 논리적인 게임 시뮬레이션 환경을 구현하기 위한 상황 표현방법을 제안한다. 그리고 상황계층구조, 사건의 정의와 분류 방법에 대해서 설명한다. 4장에서는 제안된 게임시뮬레이션 표현방법을 바탕으로 전통놀이게임 중의 하나인 바둑돌 튕기기 게임에 적용하여 시뮬레이션한다. 마지막으로 5장에서는 결론과 앞으로의 연구방향에 대해서 설명한다.

## 2. 관련 연구

컴퓨터 게임에서 시뮬레이션되는 상황은 현실세계를 추상화해서 사용자로 하여금 몰입감을 증대시키는 것이 최종적인 목표가 되고 있다. 따라서 이러한 현실감 있고, 논리적으로 연관된 시뮬레이션환경을 구축하기 위한 여러 연구가 이루어지고 있다.[8,10,15] 이러한 기반 기술을 달성하기 위해서 에이전트의 지능적인 행동에 관한 연구[11-14], 물리적 엔진의 성능향상[16,17], 분산 환경에서 각 객체들의 메시지 처리 방법 등에 대한 연구가 주를 이루고 있다.[18-21] 게임 사용자가 시뮬레이션 상황속에서 접하는 구조는 가변적인 특성을 가지고 있다. 즉 게임 내의 세계는 게임 참여자의 선택과 노력 정도에 따라 다양한 결과를 발생시킨다. 따라서 게임에서 전개되는 상황은 사용자의 선택에 지대한 영향을 미치게 된다.

게임에서 에이전트들 간의 메시지를 적절히 통제하고 처리하는 방법인 FSM 모델[18,19]은 메시지를 전달해줄 대상 에이전트나 군(group)을 지정하고 구체적인 명령을 전달해 주는 방식으로, 주위 객체들은 이 메시지의 대상이 스스로에게 해당하는 메시지만을 처리하는 방식이다.

게임 엔진에 대한 연구는 실세계에서의 문제영역에 적합한 환경에 맞도록 개발하여 물리적 문제에 대해서 효율적인 모델에 관한 연구가 진행되었고, 현재에도 보다 빠르고, 자연스러운 충돌감지에 대한 성능 개선에 대한 연구가 이루어지고 있다.[13,14,22]

온라인 게임에서의 엔진 개발은 서버와 클라이언트 부분으로 연구되고 있으며, 서버 구성 기술은 클라이언트-서버구조, 분산서버 구조, P2P(Peer to

Peer) 구조가 있다. 클라이언트는 사용자가 직접 단 일서버 또는 다중 서버중 하나의 서버에 접속해서 게임을 진행하며, 서버는 개별 클라이언트들의 행동 양상을 전체 클라이언트에게 보내는 기술이다. 따라서 서버는 각 클라이언트로부터의 얻어진 정보를 얼마나 효율적으로 처리하는가에 대한 연구와 네트워크의 부하를 적절히 분산시키는가에 대한 연구가 주종을 이루고 있다.[20,21]

현재까지의 게임구현 기술에 대한 연구는 위에서 언급한 개별 요소기술들에 대한 연구가 주를 이루고 있으며, 상황에 대한 구조적인 접근 방법에 대한 연구는 미흡한 실정이다. 즉, 어떤 형태의 게임에서도 실제 세계의 특성인 시공간적인 상황을 시뮬레이션 해야하며, 그 속에서 시나리오가 전개되는 방식이다. 따라서 이러한 시뮬레이션의 효율적인 구현과 게임 전개를 위해서는 게임상황에 대한 분석이 절실히 요구된다.

본 연구와 부분적으로 부합하는 자연현상의 시뮬레이션에 대한 연구가[15] 수행되었으나, 실세계의 자연현상은 특성상 절대적인 물리법칙에 의해서 상황이 발생되며, 사용자의 개입에 의한 상황 전개에서는 미흡한 점이 있다. 특히, 목적하는 상황 시뮬레이션과 사용자가 유발시키는 상황 시뮬레이션의 연결 관계에 대한 연구가 미흡하여 게임에 적용하기에는 분석과 설계에 있어서 어려움이 있다.

따라서 본 논문에서는 현재 게임구현기술의 많은 어려움 중에서 실세계 상황 및 진행에 대한 분석과 설계에 있어서 효율적인 게임 구현 방법을 사건에 기반한 상황 계층구조로서 효율적이고 편리한 접근이 가능함을 보이고자 한다.

### 3. 게임 시뮬레이션 환경 구축을 위한 상황계층구조

실세계 환경에 기반한 시뮬레이션 게임의 기본구조는 집단화 관계로서 표현하는 상황 계층구조로 상황, 사건, 부사건, 복합행동, 기본행동, 함수와 절차로 정의한다. 이러한 집단화 관계의 구조는 다시 3계층으로 단순화시킬 수 있다. 즉, 각 객체들의 속성들의 변화를 관리하는 속성 계층과 사건들의 인과적인 관계 및 그들 사이의 사건 발생의 전후를 관리하는 사건 계층과 실제적으로 하나 이상의 사건으로 이루어

지는 상황을 관리하는 상황 계층으로 추상화시킬 수 있다.

게임에서의 상황은 물리법칙에 근거해서 발생하며, 시공간적인 상황 속에서 인과관계에 기초하며, 상황은 단일 사건들의 집합 이루어진다는 것이다.

본 논문에서 제안하는 방법은 동적인 환경시스템에 관련된 상황의 원인과 모든 결과들의 관계를 명백하고 포괄적으로 표현할 수 있도록 아래의 상황에 중점을 둔다.

(1) 게임 상황은 사건들의 집합으로 나타나고 인과 관계에 기초해서 표현한다.

(2) 게임의 시스템적인 측면에서의 사건발생과 사용자의 입력에 의한 사건발생의 분류로 게임 내에서 발생하는 모든 사건들을 정의할 수 있다. 이러한 정의는 초기 게임제작의 설계에서 있어서 유용한 개념적인 수단을 제공하고 게임 상황을 분석할 수 있다.

(3) 게임 환경은 일련의 사건들의 집합으로 상황이 발생하며, 사건들간의 인과관계를 기반으로 다양성이 증대된다는 것이다. 다양성의 증대는 게임 참여자로 하여금 몰입감을 증대시키며, 이것은 게임이 지향하는 궁극적인 목표인 것이다.

#### 3.1 게임 시뮬레이션 환경을 구성하는 상황 계층구조

본 논문에서는 시뮬레이션게임에서의 범위를 보드게임에서 국한시켜 사건에 기초하여 표현할 수 있는 방법을 제안한다. 하나의 사건에 관련된 원인이 이전 사건의 발생에서 볼 때는 결과가 될 수 있다는 일반적이고 논리적인 개념을 응용한다. 즉 동일한 사건이 발생하더라도 사건 발생의 원인의 조건이 이전 사건의 결과 값에 따라서 새로운 사건을 발생시킬 수 있다.

현실세계와 논리적으로 유사한 시뮬레이션 게임 환경은 시공간 상황에서 전개된다. 시공간 상황의 구성은 무수한 객체들과 사건들의 연관성으로 이루어진다. 또한 상황은 더 이상 분해할 수 없는 단일사건을 기본원소로 하며, 이를 기초로 게임에 적용 가능한 시공간적인 상황계층구조는 그림 1과 같다.

그림 1은 시뮬레이션 게임 상황에 대한 일반적인 계층구조를 나타낸 것이다. 계층구조는 속성계층, 사건계층 그리고 상황계층으로 정의한다. 속성계층은 게임내에 존재하는 객체에 대한 속성에 대한 정의 및 관리이며, 속성 값은 행동에 의해서 변화를 수반

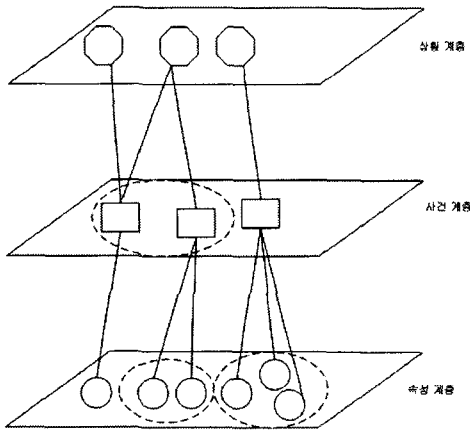


그림 1. 게임 환경 구축을 위한 상황 계층구조

하게 된다. 속성 계층은 가장 하위에 있는 계층으로 단일 노드로 표현하며, 각 객체들에 소속되어 타 객체 또는 사용자에게 의한 메시지에 의해서 제약조건의 값의 범위 내에서 변화한다. 이 계층은 각 객체들의 현재 상태와 타 객체와의 상호작용에 대한 정보도 포함한다. 특히 속성 중에서 객체의 존재유무를 결정하는 속성이 가장 중요한 사건발생의 요인이다. 왜냐하면 어떠한 상황에서는 객체의 존재유무 속성이 사건을 발생시키기 때문이다. 사건 계층은 두 개의 노드와 하나의 에지로서 단일 사건을 표현한다. 노드로 표현하는 속성과 이들의 연결인 에지는 행동으로 연결한다. 즉 행동은 두 개의 노드를 연결하여 사건을 발생시키고 일련의 사건의 발생은 상황을 유발한다. 하나의 객체내에서 속성 값의 변화뿐만 아니라 객체들간의 메시지를 이용해서 상호작용에 의한 속성 값의 변화를 사건으로 정의한다. 사건발생에서 복합 행동(complex action)은 객체의 동작이나 상태를 전이시키는 추상적인 개념이 포함된 행동이고 기본 행동(primitive action)은 모든 객체가 기본적으로 가지는 것이다. 이러한 기본 행동의 반복적이고 군집화된 추상적인 행동이 복합 행동이다. 상황 계층은 시뮬레이션 게임 진행 상태를 표현하며, 단일 사건들의 순차적, 반복적, 분기적인 연결로서 이루어지는 시간적인 흐름이다. 즉, 상황은 이러한 단일 사건들의 상호 의존성과 영향력, 즉 인과적 연결로서 발생한다. 그리고 하나의 사건도 상황으로 정의한다. 상황은 기본적으로 사건의 발생으로 게임내에서 진행되는 특정한 시점 또는 기간 동안 시공간적인 변화를 표현하는 계층이다.

### 3.2 사건의 정의와 종류

게임 시뮬레이션 내에서의 사건은 물리법칙에 기반 하여 발생하며, 상황(S)을 나타내기 위해 필요한 요소는 다음과 같이 정의한다.

$$S = \{A, A_c, E, A, E^-\}$$

여기에서,

A : 사건 발생의 원인을 제공하는 모든 속성들의 집합 =  $\{O, a_1, O, a_2, O, a_3, \dots, O, a_n\}$

A<sub>c</sub> : 속성들의 변화를 일으킬 수 있는 모든 행동들의 집합 =  $\{A_1, A_2, A_3, \dots, A_n\}$

$E \subseteq A \times A_c$  각 노드들이 에지에 의해서 다른 노드에 영향을 주는 인과성 관계를 뜻하며

E: 모든 사건들의 집합 =  $\{E^1, E^2, E^3, E^4, \dots, E^n\}$

A<sub>i</sub> ⊆ A 사건발생 대상들을 설명할 수 있는 모든 속성들이 집합

E<sup>-</sup> ⊆ E 사건들 중에서 상황을 설명할 수 있는 사건들의 집합

사건은 발생 원인과 결과에 따라서 다음과 같이 정의한다.

$$E^j = \left\{ E_i^j \mid E_i^j = f(O_{i,A}), i = 1, 2, 3, \dots, n, j = 1, 2, 3, 4, 5, 6 \right\}$$

(1)

여기에서 j는 사건의 타입을 의미하며, i는 시뮬레이션 게임 환경에서 발생하는 개개의 사건을 의미하며, O<sub>i</sub>는 객체를 나타내며, A는 객체의 속성을, f()는 객체의 함수를 의미한다. 따라서 사건은 시공간 상황에서 객체의 속성 값이 행동에 의해서 변하는 것으로 정의한다. 즉, 사건은 각 개별 객체의 행동(action)에 의해 발생한다.

$$O_i = (o-id, Evi, Ati, Rule)$$

객체는 객체의 id와 객체의 존재성, 속성, 규칙으로 구성한다. 이러한 객체의 구성은 객체의 존재를 사건의 출발점으로 고려함으로써 현실세계에서 발생하는 사건들과 개념적으로 유사성을 표현하기 위한 것이다.

A<sub>c</sub> = f<sub>j</sub>(O<sub>n,s</sub>) 행동은 객체의 상태를 변화시키며 사건발생의 수단이다. 여기에서 O<sub>s</sub>는 객체의 상태를 의미한다.

식 (1)에서 j=1,2,3,4,5,6 은 사건의 발생대상에 따른 분류이다. 게임 환경에서 발생하는 사건은 6가지로 정의하며, 이러한 구분은 게임 환경의 구조적 설계를 가능하게 하여 구현을 편리하게한다. 이러한 사

건의 분류는 객체 발생원인, 사건발생에서 속성과 객체 존재 유무에 의한 것이다. 따라서 사용자의 상호작용에 의한 사건의 발생으로 정의하는 명시적 사건과 시뮬레이션 게임 환경에서 객체의 자율적인 반응에 의한 묵시적인 사건으로 구분한다. 명시적 사건은 사용자가 키보드 또는 마우스로 특정 객체에 행동을 가해서 발생한 결과가 속성의 변화 또는 객체의 존재 유무로써 분류한다.

명시적 사건은 일반적으로 묵시적 사건의 사건발생에 원인으로 작용한다. 게임의 진행에 있어서 사용자가 특정한 메시지를 주면 이에 대응해서 게임 환경 내에 있는 객체는 사건발생의 원인정보를 수집하여 묵시적인 사건이 발생한다. 예를 들면 바둑돌 튕기기 게임에서 사용자가 흰 돌을 튕기게 되면 이에 해당하는 검은 돌은 흰 돌이 가해진 힘만큼 상하좌우 또는 대각선 방향으로 이동하게 된다.

묵시적인 사건은 게임 환경 내에서 컴퓨터가 상황을 인지하여 발생하는 사건으로 타 객체의 행동이 원인이 되어 발생하는 객체와 명시적 사건의 발생을 원인으로 발생하는 사건이다. 게임 환경적인 측면에서 묵시적 사건은 게임 환경내에 존재하는 에이전트 측면에서는 타 객체의 등장 또는 사라짐으로 사건의 행동양상이 달라진다. 따라서 객체의 등장과 사라짐은 객체의 존재유무에 기인한다.

그림 2는 시뮬레이션 환경 내에서 발생하는 묵시적 사건을 그림으로 도식화한 것이다.

그림 2의 (a)는 한 객체의 존재가 다른 객체의 속성 값을 변화시키는 사건( $E^1$ )이며, 수학적으로

$$E^1 = \{E_i^1 \mid E_i^1 = f(O_{i,A}), i = 1, 2, 3, \dots\}$$

(b)는 한 객체의 속성 값의 변화가 다른 객체의 속성 값의 변화로 유발되는 사건( $E^2$ )으로,  $E^2$ 는

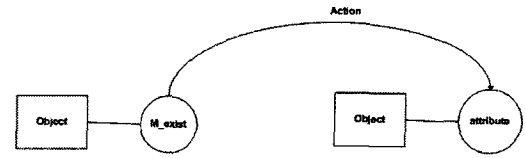
$$E^2 = \{E_i^2 \mid E_i^2 = f(O_{i,A}), i = 1, 2, 3, \dots\}$$

그림 2의 (c)는 한 객체의 속성 값의 변화가 다른 객체의 존재유무를 유발하는 사건( $E^3$ )이며, 수식으로

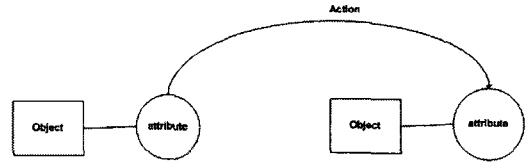
$$E^3 = \{E_i^3 \mid E_i^3 = f(O_{i,A}), i = 1, 2, 3, \dots\}$$

표현하면 나타낸다. 그림 2의 (d)는 객체의 존재유무가 다른 객체의 존재유무를 유발시키는 사건( $E^4$ )이며,

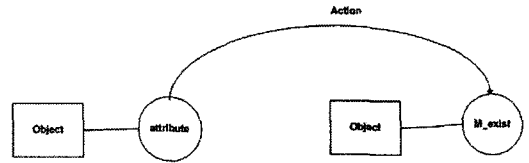
$$E^4 = \{E_i^4 \mid E_i^4 = f(O_{i,A}), i = 1, 2, 3, \dots\}$$



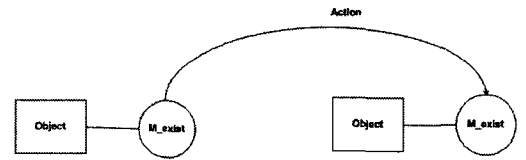
(a)  $E^1$ : 존재/속성에 의한 묵시적 사건



(b)  $E^2$ : 속성/속성에 의한 묵시적 사건

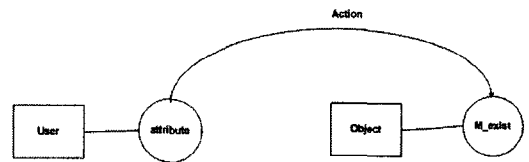


(c)  $E^3$ : 속성/존재에 의한 묵시적 사건

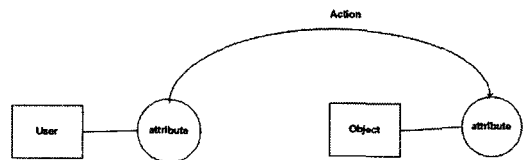


(d)  $E^4$ : 존재/존재에 의한 묵시적 사건

그림 2. 묵시적 사건의 종류



(a)  $E_i^5$ : 속성/존재에 의한 명시적 사건



(b)  $E_i^6$ : 속성/속성에 의한 명시적 사건

그림 3. 명시적 사건의 종류

그림 3은 사용자의 입력에 의해서 시뮬레이션 환경에 가해지는 명시적 사건을 나타낸 것이다. 그림 3의 (a)는 한 객체의 속성에 의해서 다른 객체의 존재가 변화하는 사건( $E^5$ )이다. 그 반대인 존재에 의해서 속성이 변화하는 사건은 명시적 사건에서는 이미 사용자가 존재함으로 사건정의에서 의미가 없다. 여기에서  $E^5$  사건은  $E^5 = \{E_i^5 \mid E_i^5 = f(O_{i,A}), i=1,2,3,\dots\}$ 로 표현한다. 그림 3의 (b)는 객체의 속성에 의해서 다른 객체의 속성의 변화를 가져오는 사건( $E^6$ )이며, 이는  $E^6 = \{E_i^6 \mid E_i^6 = f(O_{i,A}), i=1,2,3,\dots\}$ 로 나타낸다. 명시적 사건은 일반적으로 사용자의 입력에 의해서 시뮬레이션 환경에 특정 에이전트에 가해지는 메시지 형태이다. 즉 마우스 클릭, 키보드의 조작으로 행해지는 행동에 의한 것이다. 컴퓨터와 사용자간의 (User to Computer)형태의 바둑돌 튕기기 게임에서 사용자가 흰색의 바둑돌을 선택해서 검은색 바둑돌을 향하여 일정한 힘으로 흰 돌을 튕길 경우에 이것은 명시적 사건의 한 형태이다.

3.3 시뮬레이션 게임 환경에서의 명시적 사건과 묵시적 사건의 적용

게임 시뮬레이션 환경에서 존재하는 객체들은 유한하며, 게임사용자의 게임 진행에 따라 소멸, 생성된다. 본 논문에서 제안하는 방법을 설명하기 위해서 바둑돌 튕기기 게임을 예로 설명한다. 기본적인 예제 도메인은 그림 4와 같다.

사용자가 검은돌을 선택했다고 가정하자. 따라서 컴퓨터는 자연적으로 흰색 바둑돌을 선택하게 된다.

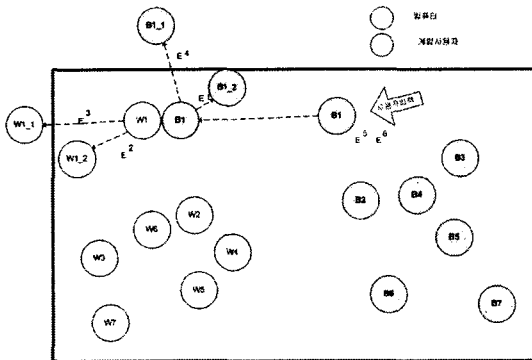


그림 4. 예제 도메인

기본적으로 7개의 바둑돌을 설정하고 게임이 진행된다고 가정한다. 먼저 게임사용자는 적군의 흰돌(W1)을 향해서 자신의 검은 바둑돌(B1: 바둑판 모서리에 가장 근접해있는 돌(W1)까지의 거리가 짧은 돌(B1))을 선택해서 공격할 것이다. 이러한 사용자에 의한 사건을 명시적 사건(그림3)으로 정의하였다. 다시 말해 명시적 사건은 키보드에 의해서 이루어지거나, 마우스에 의해서 게임사용자가 시스템에 행해진 사건을 뜻한다. 이 경우의 사건이 그림 3의 (a), (b)의 경우이다. 묵시적 사건에서 그림 2의 (c) 사건은 사용자가 검은돌(B1)을 선택해서 흰 돌(W1)을 향하여 일정한 힘으로 바둑돌을 튕겼을 경우 정해진 바둑판을 이탈(W1\_1)하는 경우이다. 즉, B1 돌의 속성값의 변화(이동)에 의해서 W1돌이 소멸되는 사건이다. 그림 2의 (B) 사건은 B1 돌의 이동으로 인하여 W1돌이 이동하여 일정하나 공간을(W1\_2)를 점유하는 사건이다. 이처럼 객체의 속성값의 변화(이동)에 의해서 다른 객체의 속성값의 변환(이동) 또는 소멸이 발생하는 사건이다. 만일 W1 돌이 W1\_1의 돌의 상태가 되면 상황은 검은돌 7개 흰돌 6개의 상태가 되며, 이것은 검은돌이 우위의 상황을 점유하고 있다고 판단할 수 있다. 이때, B1돌은 W1돌과의 충돌로 인하여 B1\_2 상태로 이동하였다고 가정한다. 여기에서 그림 2의 (D) 사건은 W1의 존재로 인하여 B1돌이 소멸되는 상태(B1\_1)이다. 그림 2의 (A) 사건은 W1의 존재로 인하여 B1돌이 B1\_2로 속성값의 변화(이동)로 새로운 공간을 점유하게 된다. 여기에서 B1 객체가 B1\_1 또는 B1\_2 상태로의 결과가 전체 게임 상황에 지대한 영향을 미치게 된다. 즉 B1\_1 이면 흰돌이 우위를 점하게 되고, B1\_2 상태이면 검은돌이 좋은 조건을 갖게 되는 것이다.

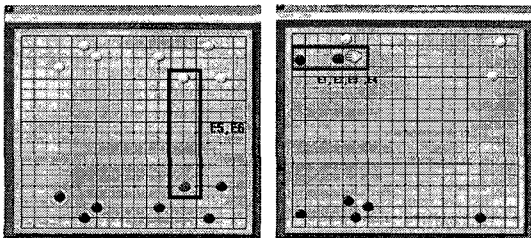
이러한 사실에 근거하여 객체의 존재 유무가 상황 전개에 주요한 요인임을 알 수 있다. 따라서 시뮬레이션 되는 환경에서 객체의 존재 유무가 가장 중요한 사건 전개의 원인임을 알 수 있다.

4. 시뮬레이션 결과

본 논문에서는 게임환경을 개발하는데 있어서 필요한 명시적 사건과 묵시적사건에 기반해서 상황 전개가 가능함을 제안하였고, 특히 객체의 존재 유무가 상황전개의 핵심적인 요인임을 설명하였다. 제안하

는 방법으로 MFC 기반의 Visual C++6.0 개발 환경에서 시뮬레이션하였다. 그림 5의 (a) 장면은 사용자가 흰돌을 제거하기 위해서 검은 돌에 의해서 흰돌의 속성의 상태가 변화 또는 존재가 결정되는 사건으로  $E^5, E^6$  사건이 발생함을 보여주고 있고, (b) 장면은 컴퓨터가 흰돌에 가하는 방향과 힘에 의해서 검은돌의 속성의 변화 및 존재가 결정되는  $E^2, E^3$ 의 사건이 발생함과 검은돌의 반작용에 의해 흰돌의 속성의 변화와 존재유무가 발생하는 사건인  $E^1, E^4$ 을 보여주는 결과이다.

시뮬레이션 결과에서 알 수 있듯이 게임내에서 발생하는 어떤 사건의 형태도 본 논문에서 정의한 사건의 한 종류이다. 따라서 게임에서 진행되는 상황은 제안하는 사건들로 세분화 할 수 있으며, 이는 객체의 존재유무 및 발생 대상에 따른 사건( $E^1, E^2, E^3, E^4, E^5, E^6$ )을 정의함으로써 효율적인 게임 설계를 위한 구조적인 접근을 가능하게 한다. 게임 발생에서 6개의 사건들의 순차적인 또는 거의 동시적인 발생으로 상황이 전개됨을 알 수 있다. 그리고 각 사건들은 바둑알들의 속성 값의 변화로 사건이 발생하며, 객체의 존재유무가 주요한 사건 발생의 또 하나의 요인인 것이다. 궁극적으로 제안하는 방법을 통하여 기존의 게임 개발에서의 문제점인 게임 전체에 대한 분석과 게임의 발생 및 진행에 대한 분석의 미흡함과, 이로 인해 게임 개발자들에게는 개발 초기에 상황에 대한 구조적인 분석 및 설계에 대한 어려움을 개선하였다. 다시 말해 본 논문에서는 이러한 문제들에 게임에서 진행되는 상황을 사건에 기반해서 분석이 가능하게 접근하였고, 이를 중심 개념으로 하여 상황 계층구조를 통하여 게임 설계의 효율성과 구현의 용이한 접근이 가능하도록 하였다.



(a) 명시적사건 (b) 묵시적 사건

그림 5. 게임 시뮬레이션

## 5. 결 론

실세계 환경 기반의 시뮬레이션 게임 환경 구축을 위한 기본 구조는 집단화관계로 표현될 수 있는 상황 계층구조로서 정의하며, 상황 계층구조를 이루는 사건을 세분화하고 이를 수학적 기호로 정의하였으며, 속성과 상황과의 관계를 정의하였다. 사용자에 의해 발생하는 명시적 사건과 시스템내에서 사용자 상호작용에 의해서 인식된 다음 행동의 양상으로 나타나는 묵시적 사건으로 분류하였다. 그리고 실제적인 현실세계의 환경을 기반으로 하는 게임 개발에 있어서 구조적인 접근 방법을 제안하였다. 첫째로, 시뮬레이션 상황은 결과적으로 무수한 객체들의 속성 값의 변화로 이루어진다. 즉, 사건으로 구성하였다. 상황은 이러한 개별 사건들의 인과적인 발생과 시간적인 흐름으로 나타나는 것이다. 따라서 개념적, 논리적 실세계 반영을 위해 사건을 정의하고 사건발생의 원인대상에 따라 세분화하여 관련성을 규명하였다. 둘째로, 사건, 객체 그리고 속성과의 관계와 행동, 상황을 수학적으로 정의하였다. 마지막으로 실세계 상황 시뮬레이션의 전개에서 객체의 존재유무가 가장 중요한 요인이며 6개의 사건으로 상황 분석이 가능함을 보였다. 궁극적으로 제안하는 방법을 통하여 기존의 게임 개발에서의 여러 문제 중에서 게임의 발생 및 진행에 대한 분석의 미흡함과, 이로 인해 게임 개발자들에게는 개발 초기에 상황에 대한 구조적인 분석 및 설계에 대한 어려움을 개선하였다. 다시 말해 본 논문에서는 게임에서 진행되는 상황을 사건에 기반해서 분석하였고, 이를 중심 개념으로 하여 상황 계층구조를 통하여 게임 설계의 효율성과 구현의 용이한 접근이 가능하도록 하였다. 그리고 제안하는 게임 상황 시뮬레이션 방법을 바둑돌 튕기기 게임에 적용하였다.

본 논문에서 제안한 사건에 기반한 상황 계층구조를 복잡한 시뮬레이션 게임, 특히 지능형 에이전트가 등장하는 게임에 적용 가능한 확장된 형태의 연구가 필요하다.

## 참 고 문 헌

[ 1 ] Lird, J.E., "Using a computer game to develop advanced AI," *IEEE Computer*, Vol. 34 No.

- 7, pp. 70-75, 2001.
- [2] Kozuki, K., Imachi, M., Ueno, M., Tsubokura, A., and Tsushima, K., "Computer game and educational system," *Proceedings of International Conference Computers in Education*, pp. 1377-1381, 2002.
- [3] Overmars, M., "Teaching computer science through game design," *The flagship magazine of the IEEE Computer Society*, Vol. 37, Issue. 4, pp. 81-83, 2004.
- [4] Nicolas Maudet, "Negotiating Dialogue Games," *Journal Autonomous Agents and Multi-Agent Systems*, Vol. 7, No. 3, pp. 229-233, 2003.
- [5] I. Badcoe, "The computer game industry: Current state of play," *Journal Virtual Reality*, Vol. 5, No. 4, pp. 204-214, 2000.
- [6] Neil C.Rowe and Tomas P.Galvin, "An Authoring System for intelligent procedural-skill Tutor," *IEEE Intelligent System & their application*, Vol. 13, No. 11, pp. 61-69, 1998.
- [7] David Franklin, "Cooperating with people: the Intelligent Classroom," *In Proceedings of the Fifteenth National Conference on Artificial Intelligence(AAAI-98)*, pp. 1-9, 1998.
- [8] Y.Matsubara, "Virtual Learning Environment for Discovery Learning and Its Application on Operator Training," *IEICE Transactions on Information & System*, E80-D, pp. 176-188, 1997.
- [9] Vijay Vasandani and T.Govindaraj, "Knowledge Organization in Intelligent Tutoring System for Diagnostic Problem Solving in Complex Dynamic Domains," *IEEE Transaction On Systems, Man, and Cybernetics*, Vol. 25, No. 7, pp. 1076-1095, 1995.
- [10] E.Ohmaye, "Simulation-Based Language Learning : An Architecture and a Multimedia Authoring Tool," Northwestern Univ, 1992.
- [11] Pattie Maes, "Artificial Life meets Entertainment: Lifelike Autonomous Agents," *Communications of the ACM*, Vol. 38, Issue. 11, pp. 108-114, 1995.
- [12] Richard Angros, Jr. W.Lewis Johnson, and Jeff Rickel, "Agents that Learn to Instruct," *In Proceedings of the Fourteen National Conference on Artificial Intelligence(AAAI-97) : Intelligent Tutoring Systems Authoring Tools*, pp. 1-8, 1997.
- [13] Ravindra Krovi, Arthur C. Graesser, and William E, "Agent Behaviors in Virtual Negotiation Environments," *IEEE Transaction On Systems, Man, and Cybernetics-Part C*, Vol. 29, Issue. 1, pp. 15-25, 1999.
- [14] Hanqiu Sun, "A Relation-Based Model for Animating Adaptive Behavior in Dynamic Environments," *IEEE Transaction On Systems, and Cybernetics-Part A: System and Humans*, Vol. 27, No. 2, pp. 1276-1291, 1997.
- [15] 박정용, 박종희, "사건 전파그래프에 기반한 동적인 자연현상의 논리적 시뮬레이션," *전자공학회논문지*, 제38권, 제4호, pp. 10-21, 2001.
- [16] Guojun Chen and Jing Zhang, "Dynamic Terrain LOD with Region Preservation in 3D Game Engine," *Lecture Notes in Computer Science*, Vol. 3942, pp. 711-715, 2006.
- [17] Falko Kuester, Gloria Brown-Simmons, Christopher Knox, and So Yamaoka, "Earth and Planetary System Science Game Engine," *Lecture Notes in Computer Science*, Vol. 3942, pp. 529-540, 2006.
- [18] Peter Linz, *An Introduction To Formal Languages and Automata Second Edition*, JONES & BARTLETT, 2001.
- [19] Alain Girault, Bilung Lee, and Edward A.Lee, "Hierarchical Finite State with Multiple Concurrency Model," *IEEE Transaction on Computer-Aided design of Integrated Circuits and System*. Vol. 18, No. 6, pp. 742-760, 1999.
- [20] Author M. V. Gubko, "Control of Organizational Systems with Network Interaction of Agents. I. A Review of Network Game Theory," *Journal Automation and Remote*



- Control*, Vol. 65, Issue. 8, pp. 1276-1291, 2004.
- [21] T. Başar and R. Srikant, "A Stackelberg Network Game with a Large Number of Followers," *Journal of Optimization Theory and Applications*, Vol. 115, No. 3, pp. 479 -490, 2002.
- [22] Chun-Yan Yu, Dong-Yi Ye, Ming-Hui Wu, and Yun-He Pan, "A new horizontal collision detection scheme for avatar with avatar in collaborative virtual environment," *Machine Learning and Cybernetics, Proceedings of 2005 International Conference*, Vol. 8, pp. 4961-4966, 2005.



박 정 응

1997년 영남대학교 전자공학과 (공학사)

1999년 경북대학교 대학원 전자공학과(공학석사)

2001년 경북대학교 대학원 전자공학과(박사수료)

2002년 3월~2006년 2월 대구산업정보대학 멀티미디어정보계열 전임강사

2006년 3월~2007년 4월 현재 대구대학교 전자공학부 누리사업초빙교수

관심분야 : 인공지능, 게임, 시뮬레이션, 교육용 시스템.