

# 비단조 추론과 귀납적 기계학습 기반 적응형 전략 게임 엔진

김 제 민<sup>†</sup> · 박 영 택<sup>††</sup>

## 요 약

요즘 사람들이 많이 즐기는 전략 게임들은 장르가 가지는 특성을 이행하지 못하고 있다. 사용자 객체의 행위에 적절히 대응하는 컴퓨터 객체의 행위를 추론해내지 못하는 물론이고 다양하게 구사되는 사용자의 전략에 대책을 마련할 수 있는 학습 능력을 갖추고 있지 못하기 때문에 사용자들은 별 다른 전략 없이 컴퓨터를 상대로 쉽게 게임을 승리할 수 있다. 이에 본 논문에서는 컴퓨터 객체에 추론 능력과 학습 능력을 적용하기 위해서 비단조 추론방식과 귀납적 기계 학습을 적용한 전략게임 인공지능 엔진을 연구한다. 본 논문에서는 다음 3가지 부분에 중점을 두고 엔진을 연구하였다. 첫째 사용자가 제어하는 객체들의 행위를 포괄적으로 모니터링하여 사용자의 객체 행위로 추상화하는 사용자 행위 모니터, 둘째 추상화된 사용자의 객체 행위에 대응하는 컴퓨터 객체들의 행위와 사용자의 전략을 학습하는 학습 엔진, 셋째 추상화되어 있는 컴퓨터 객체의 행위를 게임에 반영하는 행위 표현기를 중심으로 연구하고 있다. 특히 본 논문에서는 보다 정확하게 사용자 객체의 전략 행위를 학습하고, 사용자의 객체에 대응하는 컴퓨터 객체 행위를 만들어내기 위해서 비단조 추론과 기계 학습 기법중 하나인 귀납적 학습 방식을 적용하는 2단계의 구조를 연구하고 있다. 즉, 귀납적 학습 방법을 통해서 컴퓨터 객체가 학습한 정보를 바탕으로 비단조 추론을 이용하여 컴퓨터 객체의 행위와 전략을 결정한다. 이에 본 논문에서는 비단조 추론과 귀납적 기계 학습을 적용하여 기존 컴퓨터 객체의 행위와의 차별성을 밝혀내고, 컴퓨터 객체가 향상된 전략을 구사할 수 있게 하는 것이 주된 목표다.

## Adaptive Strategy Game Engine Using Non-monotonic Reasoning and Inductive Machine Learning

JeMin Kim<sup>†</sup> · YoungTack Park<sup>††</sup>

## ABSTRACT

Strategic games are missing special qualities of genre these days. Game engines neither reason about behaviors of computer objects nor have learning ability that can prepare countermeasure in variously command user's strategy. This paper suggests a strategic game engine that applies non-monotonic reasoning and inductive machine learning. The engine emphasizes three components - "user behavior monitor" to abstract user's objects behavior, "learning engine" to learn user's strategy, "behavior display handler" to reflect abstracted behavior of computer objects on game. Especially, this paper proposes two layered-structure to apply non-monotonic reasoning and inductive learning to make behaviors of computer objects that learns strategy behaviors of user objects exactly, and corresponds in user's objects. The engine decides actions and strategies of computer objects with created information through inductive learning. Main contribution of this paper is that computer objects command excellent strategies and reveal differentiation with behavior of existing computer objects to apply non-monotonic reasoning and inductive machine learning.

**키워드 :** 전략 게임(Strategic Game), 귀납적 기계 학습(Inductive Machine Learning), 결정 트리(Decision Tree), 추론 엔진(Inference Engine), 비단조 추론(non-monotonic Reasoning)

### 1. 서 론

전략 게임은 장애물이 존재하는 일정한 공간에서 컴퓨터가 제어하는 객체들과 사용자가 제어하는 객체들이 전투를 벌여 일정한 승리 조건을 만족시키거나 최후까지 생존하는 쪽이 게임에서 승리하는 시뮬레이션 게임 중의 하나의 장

르다. 전략 게임의 컴퓨터 객체들이 상황 판단과 학습 능력을 갖게 되면, 다양한 전략 구사가 가능하므로 컴퓨터를 상대하는 사용자는 다양한 전략을 통해 보다 재미있게 컴퓨터와 대전을 할 수 있다.

컴퓨터 객체가 자신의 행위와 전략을 결정하기 위해서는 행위와 전략 생성에 필요한 데이터가 있어야 한다. 이러한 데이터를 이용하여 컴퓨터 객체는 학습을 하게 되고, 그 결과에 따라 자신의 행위와 전략을 결정한다. 이렇게 컴퓨터 객체가 자신의 행위와 전략을 결정하기 위해 사용자의 행

\* 본 논문은 숭실대학교에 의해서 지원되었음.

<sup>†</sup> 준 회 원 : 숭실대학교 대학원 컴퓨터학과

<sup>††</sup> 정 회 원 : 숭실대학교 컴퓨터학과 교수

논문접수 : 2003년 7월 7일, 심사완료 : 2003년 12월 10일

위에 대해 반응하는 비단조 추론 방법과 귀납적 기계 학습 [4]이 필요하다. 이러한 비단조 추론방식과 귀납적 기계 학습을 통하여 생성된 데이터를 가지고 컴퓨터 객체는 사용자 객체의 행위를 추론하며, 추론된 사용자 객체의 행위를 이용하여 컴퓨터 객체는 전략을 수립하고, 생성된 전략에 따라 사용자와 대전한다.

비단조 추론방식과 귀납적 기계 학습을 적용한 기계학습 기반의 적응형 전략 게임 엔진을 구축하기 위해서 본 논문에서는 5가지 부분에 중점을 두고 엔진을 연구하였다. 행동과 전략 생성에 필요한 데이터를 수집하기 위해서, 사용자가 제어하는 객체들의 행위를 포괄적으로 모니터링하여 사용자의 객체 행위로 추상화하는 사용자 행위 모니터, 사용자 객체들의 행위에 대해 컴퓨터 객체들이 적절하게 대응할 수 있는 행위를 추론하는 추론 엔진, 추론된 데이터를 실제 게임에서 컴퓨터 객체의 움직임을 제어할 수 있는 데이터로 변환시켜 주기 위한 행위 생성기, 피드백 데이터를 입력받아 컴퓨터 객체들의 행위와 사용자의 전략을 학습하는 학습 엔진, 추상화된 컴퓨터 객체의 행위 데이터를 게임에 반영하고 그 결과를 피드백 해주는 행위 표현기를 연구하고 있다.

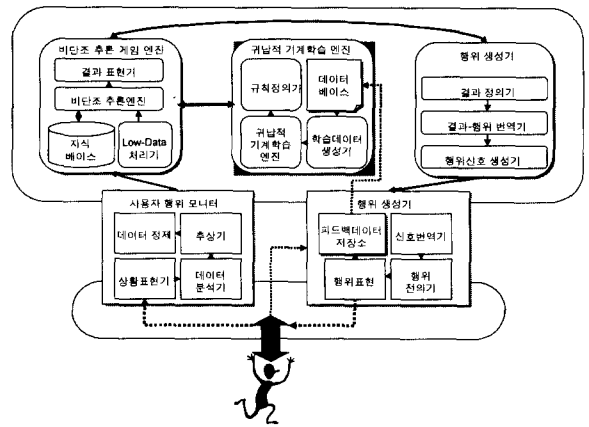
본 논문은 2장에 관련 연구를 기술하고 있으며, 3장에서는 비단조 추론방식과 귀납적 기계 학습이 적용된 전략게임 엔진에 대해 제안된 구조에 대해서 기술하고, 4장에서는 비단조 추론 게임 엔진, 5장에서는 적응형 게임 엔진, 6장에서는 사용자 모니터, 7장에서는 행위 생성기와 행위 표현기에 대해서 기술한다. 또한, 8장은 본 논문에서 제안한 비단조 추론과 기계 학습기반의 게임 인공지능 엔진에 대해 핵심인 비단조 추론 게임 엔진과 적응형 게임 엔진의 실험 결과와 학습 결과에 따라 나타나는 각 객체의 행위를 설명하도록 한다.

2. 관련 연구

컴퓨터 전략 게임은 게임을 하는 사용자에게 따라 컴퓨터와 사용자가 대전하는 1인 게임, 2인 게임, 다수 게임으로 분류할 수 있다. 이중 전략성이 많이 드러나는 게임의 형태는 컴퓨터와 사용자가 대전하는 1인 또는 2인 영합 게임(zero-sum game)인데, 영합이라는 말은 서로 상반되는 이해를 가지는 전략 게임의 경우, 한쪽(사용자 또는 컴퓨터)의 이익은 상대방의 손실을 가져오게 되어 두 경쟁자의 득실을 합하면 항상 영(zero)이 된다는 것을 의미한다. 경쟁자가 취하는 전략의 수가 유한 개수인 경우를 유한게임이라고 하고 무한인 경우를 연속게임이라 하는데, 현재 컴퓨터 게임에서는 전략의 수가 한정된 1인 또는 2인 영합 게임이 개발되고 있다. 컴퓨터 게임이 보다 다양하고 무한적인 전략을 구사하려면 게임속의 전략 학습기능이 있어야 하며, 현재 게임 속에서의 학습에 관련된 여러 가지 연구가 진행 중이다.

3. 전략게임 엔진

본 논문에서는 사용자의 행위에 따라서 컴퓨터 객체가 적절한 행동을 하고, 사용자의 전략을 학습하는 전략게임 엔진 구축을 위한 방안을 제시한다. 전략게임 엔진의 핵심은 게임 중에 컴퓨터와 대전하는 사용자 객체의 행위를 정확하게 모니터하고, 모니터 되어진 사용자 객체의 행위를 데이터로 정제한 후 사용자 객체의 행위와 전략을 파악하여, 컴퓨터 객체를 위한 행위와 전략 데이터를 보다 정확하게 구성하는데 있으며, 이를 기반으로 적절한 행위와 전략을 구사할 수 있도록 다음과 같은 구조를 제안한다.



(그림 1) 전략 게임 엔진 구조

제안하는 비단조 추론과 기계 학습 기반의 전략 게임 엔진은 사용자의 행위를 포괄적으로 모니터링하고, 컴퓨터 객체 상황과 사용자의 객체 상황을 추상적인 데이터 형태로 정제하는 사용자 행위 모니터, 정제된 데이터를 저장하고 입력된 데이터에 적합한 규칙을 찾아서 컴퓨터 객체의 행위 데이터를 추론해주는 비단조 추론 엔진, 사용자의 전략을 학습하고, 학습 결과를 저장하는 학습 엔진, 행위 생성기, 컴퓨터 객체의 다음 행위를 표현하는 행위 표현기로 구성하였다.

다음은 본 논문에서 제안하는 비단조 추론과 기계 학습 기반의 전략 게임 엔진 구조이다.

첫째, 사용자 행위 모니터는 컴퓨터를 상대로 게임을 하는 사용자의 객체에 대한 행위를 포괄적으로 모니터링하여 사용자의 객체 행위와 현재의 컴퓨터 객체 상황을 추상화한다. 이를 위해 사용자가 제어권을 가졌을 때의 사용자의 객체 행위와 상황, 그리고 컴퓨터 객체의 현재 상황을 상태-데이터 로그 파일로 만든다. 로그 파일에는 각 사용자가 게임을 하면서 나타나는 각 사용자 객체의 이동 행위, 공격 행위, 수비 행위 등을 정확히 기록하는 기능을 수행한다. 이렇게 생성한 로그 파일을 기반으로 사용자의 객체 행위와 상황을 추상화한다.

둘째, 비단조 추론 엔진은 사용자 행위 모니터에서 생성

된 추상화 데이터를 바탕으로, 사용자 객체들의 행위에 컴퓨터 객체들이 적절하게 대응할 수 있는 행위를 추론해 준다. 비단조 추론 엔진의 정확성을 극대화하기 위해서 본 논문에서는 프롤로그를 주된 모듈로 사용하고 있다. 프롤로그는 정방향 추론과 후방향 추론을 바탕으로 한 프로그래밍 언어 도구이며, 프롤로그의 바탕이 되는 후방향 추론은 A → B 즉 연역법의 IF A THEN B 에서 A가 주어졌을 때 B를 추론할 필요가 있는가를 살피고, 필요시에 A로부터 B를 얻어내는 연역법의 일종이다. 이렇게 후방향 추론을 이용해서 사용자의 객체 행위로부터 적절히 대응되는 컴퓨터 객체의 행위를 추론한다.

셋째, 귀납적 기계 학습 엔진은 행위 표현기가 만들어 준 피드백 데이터에 귀납적 기계학습 방식을 적용하여 사용자의 전략적 행위를 학습하는 기능을 수행한다. 본 논문에서는 엔트로피 개념을 활용하는 C4.5 방식을 제안하고 있다. 피드백 데이터가 사용자의 객체 행위에 대한 컴퓨터 객체 행위를 데이터로 저장하고 있으므로 각 데이터에 귀납적 방식을 적용함으로써 컴퓨터 객체를 위한 전략 데이터를 보다 정확하게 구축할 수 있게 된다.

넷째, 행위 생성기는 추론된 컴퓨터 객체의 행위들을 바탕으로 실제로 컴퓨터 객체의 행위를 결정할 수 있는 신호를 만드는 기능을 한다.

마지막으로 행위 표현기는 행위 생성기에서 만들어진 신호를 가지고 컴퓨터 객체의 행위를 표현하고, 결과물을 피드백 데이터 저장소에 저장한다. 즉 게임에서 표현된 컴퓨터 객체의 행위가 사용자와의 대전 중에 적절하지 못한 결과를 초래하면 각 객체의 행위를 단계별로 피드백 데이터 저장소에 저장하게 되며, 저장된 데이터를 바탕으로 귀납적 기계 학습 엔진은 사용자의 전략을 학습하게 된다.

#### 4. 비단조 추론 게임 엔진

본 논문에서는 보다 정확하게 사용자 객체들의 행위에 대해 컴퓨터 객체들이 적절하게 대응할 수 있는 행위를 추론하기 위하여 2단계 구조를 제안한다. 즉, 먼저 사용자 행

위 모니터로부터 정제된 데이터를 비단조 추론 엔진이 사용할 수 있는 형식으로 정형화하여, 지식 베이스에 저장한다. 다음 이렇게 정형화된 데이터를 바탕으로 컴퓨터 객체의 행위를 결정하는 사실들을 추론한다.

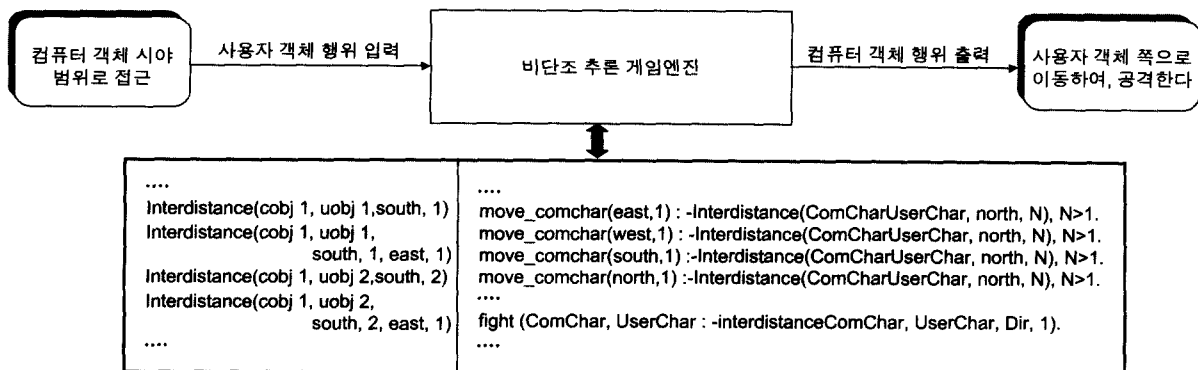
컴퓨터 객체의 행위를 결정하기 위한 1단계인 low-data 처리기는 사용자 행위 모니터가 생성한 low-data를 지식 베이스에 저장한다. 지식 베이스에 저장하기 위해서는 먼저 비단조 추론 엔진에서 추론 작업을 원활하게 하기 위해서 프롤로그에서 쓰이는 사실(fact)형식으로 변환해야한다. 그러므로 지식 베이스에는 low-data 처리기를 통해 사용자 객체의 행위와 컴퓨터 객체의 상태를 사실(fact)형태로 저장하고 있다. 또한 귀납적 기계 학습 엔진에서 학습된 사용자의 전략적 행위를 규칙으로 저장한다.

컴퓨터 객체의 행위를 결정하기 위한 2단계인 비단조 추론 단계는 지식 베이스에 사실로 담겨진 사용자 객체의 행위와 규칙으로 변환된 사용자의 전략적 행위를 바탕으로 올바른 컴퓨터 객체의 행위를 추론한다. 아래의 그림은 지식 베이스에 저장된 사실과 규칙을 바탕으로 사용자의 전략적 행위에 대한 올바른 컴퓨터 객체의 행위를 추론과정을 보여준다.

즉, 비단조 추론 게임 엔진에 “사용자의 객체가 시야에 들어오다.”라는 사용자의 객체 행위가 입력으로 들어오면 비단조 추론 게임엔진은 지식베이스에 저장된 사실과 규칙들을 이용하여 컴퓨터 객체의 행위를 추론한다. 이러한 추론 결과를 바탕으로 “사용자 객체가 있는 쪽으로 이동하여 전투 한다.”라는 컴퓨터 객체행위를 출력한다. 본 논문에서는 본격적인 추론 단계를 실행하기 위해 비단조 추론을 적용하였다. 다음은 본 논문에서 제안하는 추론엔진에 적용된 비단조 추론과 지식 기반 추론에 대해 알아보고, 이러한 개념을 바탕으로 지식베이스에 저장된 사실과 규칙들을 이용하여 컴퓨터 객체의 행위를 추론하는 과정을 설명한다.

##### 4.1 비단조 추론

일반적으로 논리에서 새로운 사실이 주어지면 그에 따라 새로운 정리가 도출되고, 이러한 정리에 따라 또 다른 정리



(그림 2) 컴퓨터 객체의 행위를 추론하는 과정

또는 사실이 나타난다. 이처럼 사실이나 정리의 수가 계속적으로 늘어나는 연역 추론과정을 단조라고 한다. 반면에 한번 밝혀진 사실이 취소되거나 다른 사실과 상충되어 거짓으로 판명되어 시간이 지남에 따라 오히려 사실이나 정리의 수가 줄어들 수 있는 데 이러한 연역 추론과정을 비단조라고 한다. 비단조 추론의 경우는 이미 추론한 사실을 다시 반복하는 것을 가능케 한다. 이것은 한번 결론 난 것을 부정해 본다거나 "What If" 분석을 통해 특정 사실이 참이 아니라면 어떠한 결론을 다시 내릴 수 있는가를 알기 위해 추론한 결론이나 규칙마다 그것을 정당화하는 근거가 됐던 다른 사실이나 규칙과의 의존 관계를 파악하고 있어야만 한다[5]. 이러한 비단조 추론법을 본 시스템의 추론 엔진에 적용하면,

위의 그림처럼 지식 베이스에

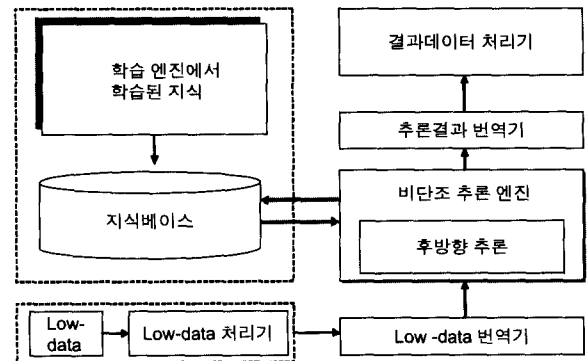
```
interdistance (cobj 1, uobj 1, south, 4)
interdistance (cobj 1, uobj 1, south, 1)
move_comchar (south,1) : ~ interdistance (ComChar,
UserChar, south, N), N>1.
fight (ComChar, UserChar) : ~ interdistance (ComChar,
UserChar, Dir, 1)
```

라는 사실과 규칙이 존재할 경우 "사용자의 객체가 서쪽 방향으로 4칸 범위에 존재한다."라는 low-data가 입력되면, 먼저 지식 베이스 내에서 입력된 low-data에 부합되는 사실인 "interdistance (cobj 1, uobj 1, south, 4)"에 대응되는 규칙인 "move\_comchar (south,1) : ~ interdistance (ComChar, UserChar, south, N), N>1."를 실행한다. 즉, "사용자의 객체가 시야에 들어오면 사용자 객체가 있는 쪽으로 이동 한다."라는 규칙이 존재할 때 컴퓨터 객체는 사용자의 객체가 시야에 들어오면 사용자 객체가 있는 쪽으로 이동 하게 된다. 또한 위의 규칙을 반복하다가 "사용자의 객체가 서쪽 방향으로 1칸 범위에 존재한다."라는 low-data가 입력 되면, 입력된 low-data에 부합되는 사실인 "interdistance (cobj 1, uobj 1, south, 1)"에 대응되는 규칙인 "fight (ComChar, UserChar) : ~ interdistance (ComChar, UserChar, Dir, 1)"을 실행한다. 즉, "사용자의 객체가 1칸 내의 시야에 들어 오면 사용자 객체와 전투 한다."라는 규칙이 존재할 때 컴퓨터 객체는 사용자의 객체가 1칸 내의 시야에 들어오면 사용자 객체와 전투를 하게 된다. 이때 컴퓨터 객체에 대한 특별한 사실이나 규칙이 존재하지 않는 한 계속 같은 행위를 반복하게 된다. 그러나 비단조 추론은 이미 추론한 사실을 나중에 반복할 수 있다. 예를 들어 적응형 게임 엔진에서 학습된 "상대적으로 전투능력이 우수하면 뒤로 후퇴하라"라는 규칙이 지식 베이스에 새로 저장이 되면 컴퓨터 객체는 일단 사용자 객체가 시야에 들어오면 사용자 객체가 있는 쪽으로 이동하지만, 사용자의 객체의 전투 능력이 자신보다 우수하므로 뒤로 후퇴하게 된다.

### 4.2 지식 기반 추론

정방향 추론은 A → B가 주어졌을 때 A가 작업 메모리에 있으면 B를 실행 시킨다. 후방향 추론은 A → B 즉 IF A THEN B에서 A가 주어져 있을 때 B를 추론할 필요가 있는가를 살피고, 필요시에 A로부터 B를 얻어내는 연역법의 일종이다. 후방향 추론의 사용하는 이유는 수많은 규칙을 모두 수행하지 않고 조건을 만족하는 규칙을 찾아서 수행하기 위한 추론의 효율성 때문이다. 이러한 후방향 추론을 본 시스템의 추론 엔진에 적용하면, 게임 상에서 컴퓨터 객체를 동쪽으로 한 칸 이동시키는 상황이 필요할 경우 추론 엔진은 지식 베이스에 저장된 사실과 규칙들을 탐색하게 되고, 컴퓨터 객체를 이동시키는 규칙을 찾는다. 즉 컴퓨터 객체를 이동시키기 위해 지식 베이스에 있는 모든 규칙을 실행해 보는 것이 아니라, 적합한 규칙만을 찾아내어 실행한다. 본 논문에서는 정방향 추론과 후방향 추론 기법을 사용하는 추론 엔진을 구축하기 위해서 프롤로그를 주된 모듈로 사용한다.

다음은 본 논문에서 제안하는 비단조 추론 게임 엔진 구조이다.



(그림 3) 비단조 추론 게임 엔진

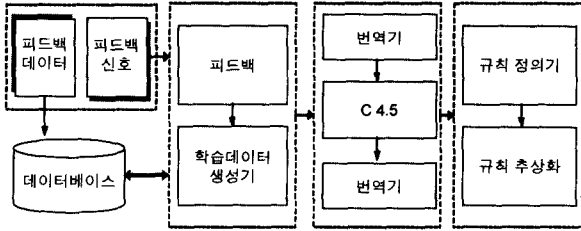
사용자 행위 모니터로부터 정제된 low-data들은 low-data 처리기에 의해 차례로 처리 되어진 후 low-data 번역기에 의해 사실(fact)로 변환되어 비단조 추론엔진에 넘겨진다. 비단조 추론 엔진은 넘겨받은 사실을 지식 베이스 내에 저장하고, 사실을 만족하는 규칙을 선택하여 결과 데이터를 추론한다. 추론된 데이터는 결과 번역기에 의해 행위 생성기에서 사용할 수 있는 형식으로 변환된 후, 결과 데이터 처리기를 통해서 행위 생성기로 전송된다.

### 5. 적응형 게임 엔진

일반적으로 비단조 추론 엔진에 의해 추론된 컴퓨터 객체의 행위는 불완전(incomplete), 부정확(incorrect)한 특성을 가지게 된다. 따라서 이러한 문제를 보완하여 보다 완벽하고 정확히 컴퓨터 객체의 행위를 추론하기 위해서 본 논

문에서는 사용자의 전략적 행위를 학습하는 적응형 게임 엔진 엔진을 이용한다.

다음은 본 논문에서 제안하는 적응형 게임 엔진 구조이다.



(그림 4) 적응형 게임 엔진

적응형 게임 엔진은 게임 상에서 컴퓨터 객체의 행위가 자신에게 부적합한 결과를 야기할 경우, 이러한 부적합한 행위를 도출한 데이터를 행위 표현기로부터 전달받아 데이터베이스에 새로운 데이터로 저장하게 된다. 이렇게 저장된 데이터는 기존에 저장되었던 데이터들과 함께 학습 데이터 생성기에 의해 학습될 수 있는 데이터 형식으로 변환 된다. 이렇게 변환된 학습 데이터를 바탕으로 학습을 실행하게 되고, 학습 결과는 규칙-정의기에 의해 새로운 규칙으로 정의되어 지식 베이스에 저장된다.

본 논문에서는 사용자의 전략적 행위를 학습하기 위해서 엔트로피 개념을 활용하는 C4.5 귀납적 기계학습 알고리즘을 이용한다[9, 10]. 본 논문에서 사용하는 C4.5는 Ross Quinlan이 분류모델(Classification model)로서 데이터베이스에 수집된 트레이닝 데이터를 대상으로, 각 트레이닝 데이터를 대표하는 특성을 발견하고 분석한다. 분석된 정보는 영역을 분류하는 모델을 구성하고 이 모델은 각각의 속성 값에 따라 결정트리를 생성한다. 이때 C4.5는 결정트리를 생성하기 위해서 정보 이론(Information Theory)을 바탕으로 하는 gain값을 사용하는데 gain값을 구하는 식은 다음과 같다. 여기서 S는 전체 트레이닝 데이터 집합이며, A는 데이터의 각각의 속성을 나타내고, S<sub>v</sub>는 속성의 값을 나타낸다.

$$Entropy \equiv \sum_{i=1}^c -p_i \log_2 p_i$$

$$Gain(S, A) \equiv Entropy(S) - \sum_{v \in Values(A)} \frac{|S_v|}{|S|} Entropy(S_v)$$

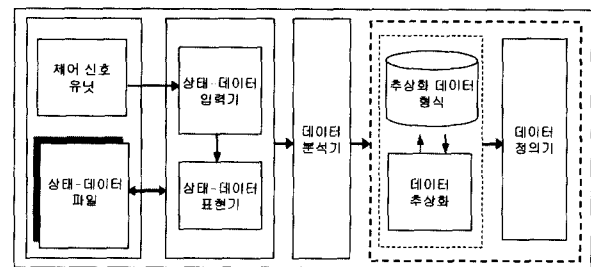
C4.5를 이용하여 생성되는 결정트리는 결정 변수를 나타내는 내부 노드와 분류결과를 나타내는 단말 노드 그리고 이들을 서로 연결하는 선(edge)으로 이루어져 있다. 이러한 트리 생성은 사용자의 전략을 파악하는데 중요한 속성 순으로 생성되며, 지식의 형태로 인식되어 지식 베이스에 저장한다. 즉 C4.5는 결정트리를 이용하여 사용자의 전략에 부합되는 중요한 속성 값에 대한 일정한 규칙을 생성한다. 본 논문에서 많은 학습방법 중에서 귀납적 기계학습 알고리즘인 C4.5를 이용하는 이유는 컴퓨터와 사용자간의 상호작용을 하면 할수록 이전보다 더욱 효과적인 학습을 처리하는 학습 시스템이 가능

하기 때문이다. 이처럼 학습된 사용자의 전략적 행위는 규칙 정의(rule define)과정을 거쳐서 지식 베이스에 저장되어 컴퓨터 객체의 행위를 추론하는 기능을 수행한다.

## 6. 사용자 행위 모니터

전략게임 인공지능 엔진을 구축하기 위해서 가장 먼저 수행되어야하는 것은 사용자의 객체가 취하는 행위들과 현재의 컴퓨터 객체의 상태를 포괄적으로 탐지하는 것이다. 그래서 본 논문에서는 사용자 행위 모니터를 이용하여 사용자 객체의 행위와 컴퓨터 객체의 상태를 모니터하고, 각 객체의 위치와 속성, 대치 상황 등을 추출하여 로그 파일로 만든 후, 분석한다.

다음은 본 논문에서 제안하는 사용자 모니터의 구조다.



(그림 5) 사용자 모니터

본 논문에서는 컴퓨터와 게임을 하는 사용자 객체의 모든 행위를 탐지하여 저장하는 방법으로 제어 신호 장치(control signal unit)와 상태-자료 파일(state-data file)을 사용하였다. 제어 신호 장치(control signal unit)는 일반적으로 게임 중에 게임 제어권이 컴퓨터에 넘어갔을 때 제어 신호를 통해서 임시로 저장되는 매우 작은 텍스트 파일의 형태이다. 게임의 제어가 컴퓨터에 넘어오면, 이렇게 만들어진 임시 파일을 통해서 게임에서 보여줬던 사용자 객체의 행위와 컴퓨터 객체의 상태에 대한 정보들을 저장할 수 있다. 또한 게임 시스템 내에서는 사용자와 컴퓨터 객체에 대한 정보가 들어올 경우, 사용자와 컴퓨터 객체의 정보 변화에 대한 탐지를 수행하는 중요한 역할을 한다.

제어 신호 장치 및 상태-자료 파일을 통해서 얻어진 정보는 모니터가 쉽게 사용자 객체 행위와 컴퓨터 객체의 상태에 대한 형식화 된 기록 정보를 구축하기 위해서 필요한 정보만을 찾아내는 분석 작업을 하게 되며, 분석된 데이터들은 데이터 추상화를 통하여 일정한 품으로 벡터 구조에 저장되며, 저장된 추상 데이터들은 비단조 추론 엔진에서 사용할 수 있는 low-data로 정제 된다. 이렇게 정제된 정보를 통해서 정보를 이용하는 비단조 추론 엔진이 사용자 객체들의 행위에 컴퓨터 객체들이 적절하게 대응할 수 있는 행위를 추론한다. 이처럼, 본 논문에서 사용자 행위 모니터는 전략 게임 내에서 일어나는 사용자의 모든 행위와 컴퓨터 객체의 상태를 기록하는 것

뿐만 아니라, 기록된 데이터를 비단조 추론 엔진이 사용할 수 있도록 정제하는 과정을 포함한다.

### 7. 행위 생성기와 행위 표현기

비단조 추론 엔진에서 추론된 컴퓨터의 객체의 행위를 실제 컴퓨터 게임 상에 표현하기 위해서는 규칙 데이터에서 추출된 결과를 컴퓨터 객체의 움직임을 제어하는 데이터로 변환시켜야 한다. 본 논문에서는 비단조 추론 게임 엔진에서 추론된 데이터를 실제 게임에서 컴퓨터 객체의 움직임을 제어할 수 있는 데이터로 변환시켜 주기 위한 행위 생성기를 제안한다. 행위 생성기는 2단계 구조로 구성된다. 먼저 추론 엔진에서 추론된 데이터는 데이터 번역기에 의해서 게임 프로그램에서 읽어 들일 수 있는 데이터로 변환된다. 예를 들어 컴퓨터 객체를 동쪽으로 한 칸 움직이려는 추론 결과는 데이터 번역기에 의해 다음과 같이 변환된다.

```

move_comchar(east,1) => if (action00 = move_comchar)
                        dir = "east"
                        dis = 1
    
```

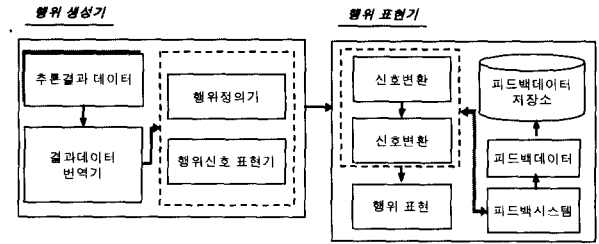
이렇게 변환된 데이터는 행위 정의기에 의해 행위 표현기가 실제 컴퓨터 객체를 제어할 수 있는 행위 신호를 만들어주게 된다. 행위 표현기는 행위 생성기에서 생성한 행위 신호를 바탕으로 컴퓨터 모니터에 컴퓨터 객체의 전략적 행위 표현해준다.

일반적으로 비단조 추론 게임 엔진에 의해 추론된 컴퓨터 객체의 행위는 불완전(incomplete), 부정확(incorrect)한 특성을 가지게 된다. 따라서 이러한 문제를 보완하기 위해 본 논문에서는 데이터 피드백 시스템을 이용한다. 이는 피드백 데이터 저장소를 이용하여 추론된 컴퓨터 객체의 행위를 분석하여 실패(failure)로 간주되는 경우를 이용하여, 학습 엔진의 학습 데이터로 제공하는 것이다. 게임에서 보여 지는 컴퓨터 객체의 행위에 따라서 다음과 같은 종류를 실패(failure)로 정의한다. 첫째, 컴퓨터 객체의 행위가 일어난 후, 사용자가 게임을 제어하는 동안 컴퓨터 객체가 게임에서 아웃 되거나, 아웃 될 상황을 만나는 경우와 둘째 컴퓨터 객체의 어떠한 전략적 행위가 기대 이하의 효과를 보이는 경우이다.

본 논문에서는 컴퓨터 객체의 행위를 정확하게 만들어내기 위하여 실패(failure)가 발생한 경우 컴퓨터 객체의 행위를 불완전(incomplete), 부정확(incorrect)하게 만든 데이터를 추출한 다음 피드백 데이터 저장소에 저장하여 적응형 학습 엔진에서 학습을 통해 정확한 데이터로 정제한다. 즉 피드백 시스템을 통해 다음번에는 사용자 객체의 전략적 행위에 대해 보다 정확한 컴퓨터 객체의 행위를 추론하여 표현하는 것이다.

다음은 본 논문에서 제안하는 행위 생성기와 행위 표현

기의 연결된 구조이다.



(그림 6) 행위 생성기와 행위 표현기

### 8. 실험 및 결과

본 논문에서 제안하는 비단조 추론과 귀납적 학습이 적용된 전략 게임 엔진은 게임이 진행되는 중에 사용자 객체의 행위에 따라서 컴퓨터 객체가 적절한 행동하고, 사용자의 전략을 추론함으로써 게임 사용자가 보다 재밌게 게임을 즐기도록 하는 기능을 제공한다. 이와 같은 기능은 전략 게임이 실행되면서 다양하게 구사되는 사용자의 전략적 행위를 파악하고, 이를 기반으로 컴퓨터 객체에게 필요한 행위를 추론해줌으로써 이루어진다. 이러한 사용자의 전략적 행위의 파악과 컴퓨터 객체를 위한 행위 추론은 그에 따른 추론 엔진과 학습 엔진이 정확하게 구축되어야 가능하다. 그래서 본 논문에서 제안하는 비단조 추론과 귀납적 학습이 적용된 전략 게임 엔진의 성능을 평가하기 위해서 사용자 객체의 전략적 행위에 따른 컴퓨터 객체의 행위 추론에 대한 정확성 평가와 사용자 객체의 전략적 행위를 학습하는 것과 그 학습 결과를 컴퓨터 객체의 행위에 적용했을 때의 대한 정확도 평가를 수행하도록 한다.

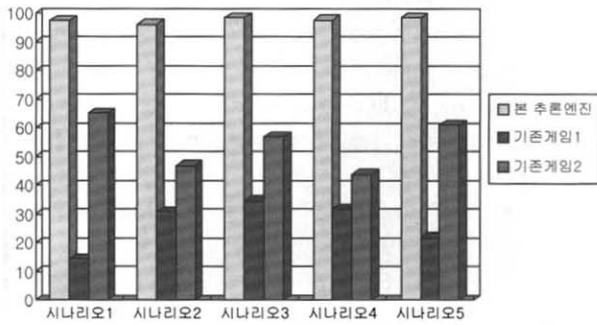
<표 1> 실험에 활용된 자료

	Low-data의 수	시나리오의 수	컴퓨터의 객체 행위의 종류	사용 규칙의 수	비교대상
실험 1	20	5	50	80	기존게임 1 기존게임 2
실험 2	2	2	50	20	기존게임 1 기존게임 2

- 실험을 평가하기 위한 척도로 정확도의 평균을 이용.
- 기존게임 1은 파이널 텍세스4, 기존게임 2는 삼국지 조조전을 사용.

첫째, 사용자 객체의 전략적 행위에 따른 컴퓨터 객체의 행위 추론에 대한 정확성 평가이다. 본 논문에서는 컴퓨터 객체의 행위를 추론하기 위해서 지식 베이스를 구축하여 프롤로그를 주된 모듈로 사용하는 비단조 추론 게임엔진을 구축하였다. 구축된 추론엔진의 정확성을 평가하기 위해서는 먼저 사용자 객체의 행위가 지식 베이스에 사실(fact)로 잘 구축되었는지에 대한 실험이 필요하다. 또한 지식 베이스에 사실과 규칙을 바탕으로 컴퓨터 객체의 행위를 얼마나 정확히 추론해 내는지에 대한 실험이 필요하다.

다음의 그림은 본 논문에서 제안하는 비단조 추론 엔진이 어느 정도 정확하게 시나리오에 따른 컴퓨터 객체의 행위를 추론해주는지 그래프로 표현한 것이다. 또한 같은 시나리오를 기존의 전략 게임에 적용시켰을 경우 얼마나 정확히 시나리오에 따른 행위를 컴퓨터 객체의 행위로 보여주는지 그래프로 표현하였다.



(그림 7) 컴퓨터 객체의 행위 추론 실험

먼저 비단조 추론 게임 엔진이 적용된 전략게임 인공지능 엔진의 경우, 평균 95%의 정확도를 나타내었다. 반면에 기존게임 1과 기존게임 2는 각 시나리오가 요구하는 컴퓨터 객체의 행위를 제대로 표현하지 못했다. 실험 결과에 따르면 본 논문에서 제안하는 전략게임 엔진에 비단조 추론 엔진을 적용할 경우 적용되지 않은 기존게임 보다 컴퓨터 객체의 행위를 더욱 정확히 유추해 주었다.

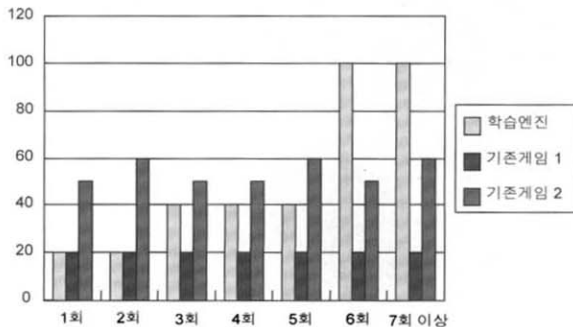
둘째, 사용자 객체의 전략적 행위를 학습하는 것과 그 학습 결과를 컴퓨터 객체의 행위에 적용했을 때 대한 정확도 평가이다. 본 논문에서는 사용자 객체의 전략적 행위를 학습하기 위하여 적응형 게임 엔진을 구축하였다. 본 논문에서는 컴퓨터 객체의 행위를 실패로 만드는 두 가지 시나리오를 만들어서 컴퓨터 객체의 전략적 행위를 유도할 수 있는 규칙이 학습 되는지를 기존 게임과 비교하여 실험해 보았다. 먼저 첫 번째 상황은 컴퓨터 객체보다 상대적으로 전투적 능력이 우수한 사용자의 객체를 컴퓨터 객체와 대치

시킨 경우인데, 이 경우 학습이 이루어지지 않은 상황에서 기존의 추론된 행위를 입력받은 컴퓨터 객체는 자신보다 월등한 능력을 가진 사용자 객체와 전투를 벌이다 게임에서 아웃되는 실패를 겪는다. 두 번째 상황은 컴퓨터 객체보다 상대적으로 전투 능력이 떨어지는 사용자의 객체가 주변에 동료 객체를 매복시킨 후 컴퓨터 객체와 대치시킨 경우인데, 이 경우 역시 학습이 이루어지지 않은 상황에서 기존의 추론된 행위를 입력받은 컴퓨터 객체는 자신보다 낮은 능력을 가진 사용자 객체의 유인 작전에 걸려 협공을 받고 게임에서 아웃되는 실패를 겪는다.

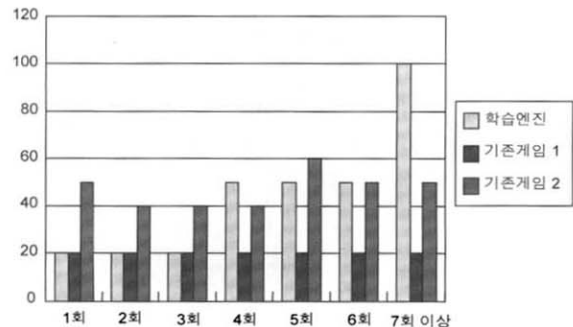
적응형 게임 엔진이 두 가지 사용자의 전략적 행위를 어느 정도 정확하게 학습하는지 평가하기 위하여 같은 상황을 7회 이상 반복 실행하여, 컴퓨터 객체의 행위가 실패하는 시간을 측정해본 결과, (그림 8)과 같은 결과를 보였다

먼저 상황 1의 경우, 기존 게임 1의 경우는 전혀 사용자의 전략적 행위에 대해 대처를 하지 못한다. 또한 기존 게임2의 경우 어느 정도 사용자 객체의 행위에 대해 대처할 수 있는 내부 프로그램은 존재하는 것으로 보이지만, 사용자 객체의 전략적인 행위를 학습하여 적절한 행위를 도출하지는 못한다는 결과를 얻었다. 반면에 적응형 게임 엔진이 장착된 전략 게임 인공지능 엔진의 경우, 같은 상황을 반복시켰을 경우 사용자의 전략적 행위를 학습하여 새로운 규칙을 생성해서 적절히 대처할 수 있는 컴퓨터 객체의 행위를 표현되었음을 알 수 있다. 상황 1에서 학습 엔진이 학습한 규칙은 “컴퓨터 객체를 이동 상태에서 멈춘다.”와 “컴퓨터 객체를 자신의 동료 객체가 있는 방향으로 한 칸 후퇴시킨다.”이다.

상황 2의 경우도 상황 1과 비슷한 결과를 냈다. 즉 기존 게임 1과 기존 게임 2의 경우 사용자의 전략적 행위(유인 작전)에 제대로 대처하지 못한 반면, 적응형 게임 엔진이 장착된 전략 게임 인공지능 엔진의 경우, 같은 상황을 반복시켰을 경우 사용자의 전략적 행위를 학습하여 새로운 규칙을 생성해서 적절히 대처할 수 있는 컴퓨터 객체의 행위를 표현되었음을 알 수 있다. 상황 2에서 학습 엔진이 학습한 규칙은 “컴퓨터 객체를 이동 상태에서 멈춘다.”와 “다른



(a) 상황 1



(b) 상황 2

(그림 8) 사용자 객체의 전략적 행위를 학습하는 실험

동료 객체를 자신이 있는 방향으로 한 칸 전진시킨다.”이다.

위 두 가지 실험 결과를 통하여, 본 논문에서 제안한 전략게임 인공지능 엔진이 사용자 객체의 전략적 행위를 학습한 후, 이를 기반으로 정확한 컴퓨터 객체의 행위를 추론하는 기능을 실행한다는 것이 정확하다는 것을 확인 할 수 있었다. 이러한 실험 결과를 통하여 추론엔진과 학습엔진을 탑재한 전략 게임이 더욱 우수한 컴퓨터 객체의 전략적 행위를 보여줄 수 있음을 알 수 있다.

### 9. 결 론

비단조 추론과 귀납적 학습이 적용된 전략 게임 엔진은 사용자 객체의 행위를 모니터 하면서, 그 행위를 추상적인 데이터로 정제하여 지식 베이스를 구축한다. 구축된 지식 베이스의 사실과 규칙을 바탕으로 사용자 객체의 행위에 대응하는 컴퓨터 객체의 행위를 추론하고, 게임에 이러한 행위를 표현한다. 만약 적절한 행위가 추론되지 못할 경우 피드백 시스템을 이용하여 사용자의 행위를 분석하고 학습하여 정확한 컴퓨터 객체의 행위를 이끌어 낸다. 본 논문에서는 보다 정확하게 사용자 객체들의 행위에 대해 컴퓨터 객체들이 적절하게 대응할 수 있는 행위를 추론하기 위하여 2단계 구조를 제안하였다. 이러한 2단계 구조는 사용자 행위 모니터로부터 정제된 데이터를 추론 엔진이 사용할 수 있는 사실(fact)형식으로 정형화하여, 지식 베이스에 저장한다. 다음 이렇게 저장된 사실을 바탕으로 비단조 추론 게임 엔진이 추론함으로써 보다 정확한 컴퓨터 객체의 행위를 결정할 수 있다. 또한 비단조 추론 게임 엔진에 의해 추론된 컴퓨터 객체의 행위가 불완전(incomplete), 부정확(incorrect)한 특성을 가지는 문제를 보완하여 보다 완벽하고 정확히 컴퓨터 객체의 행위를 추론하기 위해서 적용형 게임 엔진을 구축하였다. 본 논문에서 제안하고 있는 비단조 추론과 귀납적 학습이 적용된 전략 게임 엔진은 사용자 객체의 전략적 행위를 학습한 후, 이를 기반으로 정확한 컴퓨터 객체의 행위를 추론하는 기능을 실행한다는 것이 정확하다는 것을 실험을 통해 확인 할 수 있었다. 이러한 실험 결과를 통하여 추론엔진과 학습엔진을 탑재한 전략 게임이 더욱 우수한 컴퓨터 객체의 전략적 행위를 보여줄 수 있음을 알 수 있다.

### 참 고 문 헌

[1] welcome to game AI, <http://www.gameai.com/ai.html>.  
 [2] Brodley, C. E. and utgoff, P. E., Multivariate decision tree. Machine Learning, 19, pp.45-77, 1995.

[3] PRYOR, K. Clicker training for dogs. Sunshine Books, Inc., Waltham, MA. 1999.  
 [4] Mitchell, K. Machine learning. McGraw Hill, New York, NY, 1997.  
 [5] Thomas, D. Artificial Intelligence, Addison-Wesley Pte. Ltd., pp.187-212, 1995.  
 [6] black&white, <http://www.digipen.edu/~tfolsom/cs381/lectures/black&white>.  
 [7] Combinatorial Game Theory, [http://www.aistudy.co.kr/game/com\\_game\\_theory.htm](http://www.aistudy.co.kr/game/com_game_theory.htm).  
 [8] Kenneth A. De Jong and Alan C. Schultz, Using experience-Based Learning in Game playing, George Mason University, 1988.  
 [9] JR. Quinlan, "Induction of Decision Tree," Machine Learning, pp.81-106, 1986.  
 [10] JR. Quinlan, "C4.5 Programs for Machine Learning," San Mateo, CA : Morgan, Kaufaman, 1993.  
 [11] Elwyn Berlekamp and David Wolfe, "Mathematical Go: Chilling Gets the Last Point," A K Peters Ltd. 1994.  
 [12] Baird, Douglas G., Robert H. Gertner and Randal C. Picker, Game Theory and the Law, Cambridge Mass. : Harvard University Press, 1994.  
 [13] Tempest, <http://www.softmax.co.kr/tempest/english>.



#### 김 제 민

e-mail : kimjemins@hotmail.com  
 2001년 숭실대학교 컴퓨터학과(학사)  
 2004년 숭실대학교 대학원 컴퓨터학과(석사)  
 2004년~현재 숭실대학교 대학원 컴퓨터학과 박사 과정

관심분야 : 게임 AI, 시멘틱 웹, 유비쿼터스



#### 박 영 택

e-mail : park@computing.soongsil.ac.kr  
 1978년 서울대학교 전자공학과(학사)  
 1980년 KAIST 전산학(석사)  
 1992년 Univ. of Illinois at Urbana-Champaign(박사)  
 1981년~현재 숭실대학교 컴퓨터학과 교수

관심분야 : 시멘틱 웹, 에이전트, 전문가 시스템, 유비쿼터스