

퍼지논리를 이용한 윗놀이 인공지능 플레이어 연구

정성욱* · 김기년**

Study on an Artificial Intelligence Player of the Yutnori Game Using the Fuzzy Logic

Chung Sungwook · Kim Kinyun

〈Abstract〉

Recently, the Go game has been performed between the 'AlphaGo' of the DeepMind and Lee Sedol, a famous professional Go-player of Korea, which leads to arise a lot of interests in the AI (Artificial Intelligence) research area. Based on the Fuzzy logic of the AI, we have also developed another game's AI, i.e., the Yutnori game, one of Korean traditional board games. However, it is not easy and simple to consider all the cases of the Yutnori game since it is a non-perfect information game in terms of the AI. Thus, we have developed the Fuzzy-logic-based AI which tries to simulate humans' selections, meaning that the suggested AI has focused on the humans' choices depending on diverse situations in the Yutnori. With our extensive simulations using the suggested Yutnori AI, we have analyzed its performances with respect to 10 Yutnori situations among various scenarios. In conclusion, our suggested AI have demonstrated that 6 out of 10 situations are exactly same with the humans' choices and the rest 4 cases are also similar to that of human's, which reveals that our Fuzz-logic-based Yutnori AI can effectively simulate human's choices.

Key Words : AI, Fuzzy Logic, AI-Enabled Game Design, Yutnori

I. 서론

인공지능은 최근 컴퓨터 공학에서 각광받고 있는 분야 중의 하나이다. 최근 인공지능 내의 이슈는 구글의 바둑 인공지능 프로그램인 '알파고'이다. 알파고는 2016년 9월 9일부터 15일까지 5회에 걸친 바둑 대결 끝에 한국의 프로 바둑기사 이세돌을 상대로 4

승 1패로 승리하였다. 이는 1997년 IBM이 슈퍼컴퓨터인 딥블루를 사용하여 그 당시 세계 체스 챔피언 Garry Kasparov를 이긴 후 장기간 동안 인공지능이 프로바둑기사를 이기기는 어렵다는 기존의 생각을 깼다[1]. 장기간 동안 인공지능이 바둑을 정복하지 못한 이유는 바둑은 체스와 달리 게임의 경우의 수가 너무 많기 때문이었다. (체스의 경우 Breadth \approx 35, Depth \approx 80을 갖고 바둑의 경우 Breadth \approx 250, Depth \approx 150이다). 하지만 구글은 딥러닝(Deep

* 창원대학교 컴퓨터공학과 조교수

** 창원대학교 컴퓨터공학과 학사

Learning)과 몬테카를로 서치(Montecarlo Search)를 사용하여 인간이 오랫동안 축적해온 바둑 기보를 일련의 정형화 과정을 통해서 알파고에게 학습시켰고 결국 기존의 통념을 깼다[1]. 이는 인공지능이 여러 분야에 적용될 수 있음을 시사했고 다양한 분야의 인공지능 연구의 필요성을 가져왔다. 그래서 본 논문은 윗놀이를 통하여 인공지능을 연구하고자 하였다.

이 연구의 목적은 바둑과 같은 완전정보 게임(Perfect Information Game)뿐만 아니라 윗놀이와 같은 불완전 정보 게임(Non-perfect Information Game)의 인공지능을 개발하는 것에 있다. 이러한 불완전 정보 게임의 문제점은 완전 정보 게임에 비해 비교적 복잡도가 높다는 것이고 이를 해결하기 위해 여러 가지 알고리즘이 있지만 이 연구에서는 디아블로2와 같은 주로 RPG게임에 쓰였던 퍼지 논리(Fuzzy Logic)를 통해 인공지능을 구현하였다[2].

그 이유는 최근 인기가 있는 딥러닝 방식은 윗놀이 인공지능을 구현하는 것에 있어 두 가지 단점을 가지고 있기 때문이었다. 첫 번째 단점은 딥러닝은 퍼지논리에 비해서 대규모의 데이터를 저장하고 처리하고 연산할 수 있는 저장장치 혹은 연산장치가 필요하고 퍼지논리에 비해 구현이 난해하기 때문이었다.(한 예로 알파고는 1920개의 CPU와 280개의 GPU를 사용하였다[3]) 두 번째 이유는 윗놀이와 같은 게임은 알파고에서 컴퓨터의 학습 시 사용되었던 '기보'와 같은 정형화 된 다량의 데이터가 없었기 때문이었다[1][3][4].

이 연구의 인공지능의 구현방법은 우선 각각의 상황을 인식하는 퍼지 함수(Fuzzy Function)를 구현하였다. 그리고 이를 통해 남은 거리, 상대편 말과의 관계, 지름길로 갈 수 있는 확률, 상대편 말을 잡을 수 있는 확률을 멤버십 치(Membership Grade)라고 하는 0과 1사이의 수치로 변화시켜 최종적으로 퍼지

연산(Fuzzy Operation)을 통해 가장 최적 값을 가진 말이 선택되도록 구현하였다. 그리고 사용자 유저 인터페이스를 제작하여 실제 3종류의 플레이어간의 게임 결과를 관찰하였다.

이 연구를 통해서 예상되는 효과는 지금까지 윗놀이 인공지능에 관한 연구가 많이 없었기 때문에 본 논문이 향후 다른 윗놀이 인공지능 연구나 다른 주제의 인공지능의 연구 시에도 도움이 될 것 이라고 생각한다.

본 논문의 순서는 2장에서 국내외 게임 인공지능 연구 동향을 소개하고 3장에서는 윗놀이 인공지능 연구의 필요성과 Fuzzy Logic에 대해서 설명한다. 4장에서는 연구에 사용된 시스템 구성과 어떻게 윗놀이 게임에 Fuzzy Logic을 적용하였는가에 대해서 설명한다. 5장에서는 실제 구현 모델을 제시하고 인공지능과 인간간의 게임 결과에 따른 성능평가를 제시하고 분석하고 향후 개선방향에 대해서 제안하고 6장에서 이 연구의 결론을 제시한다.

II. 게임관련 인공지능 연구 동향

2.1 딥 블루(Deep Blue)

딥 블루(Deep Blue)는 IBM이 체스 인공지능 개발을 위해 만든 컴퓨터로 1997년 러시아의 체스 세계 챔피언 Garry Kasparov와의 대결에서 승리했다. 딥 블루의 인공지능 방식은 가능한 모든 경우를 조사하는 방식으로 이를 위해 대규모의 병렬처리 능력을 위해 30개의 노드로 구성된 RS/6000 SP기반 컴퓨터를 사용했고 480개의 전용 VLSI를 사용했다. 1초당 2억 개의 수를 계산할 수 있으며 12수 앞까지 모든 경우의 수를 계산할 수 있다[5-6].

2.2 알파 고(Alpha Go)

알파고(Alpha Go)는 구글에 인수된 딥-마인드가 개발한 바둑 인공지능(AI)프로그램이다. 2016년 세계 최상위급 바둑 프로기사 이세돌과의 경기에서 4승1패로 승리했다. 네트워크를 사용한 분산컴퓨팅을 사용하며 1920개의 CPU와 280개의 GPU를 사용하였다. 알파고에 사용된 알고리즘은 최적의 위치를 계산하는 정책망(Policy Network)과 승률을 계산하는 가치망(Value Network)을 이용하여 몬테카를로 서치를 사용하여 최적의 위치를 정하게 된다[1-3].

2.3 스타크래프트(Starcraft)

딥-마인드는 단순히 보드게임뿐만 아니라 스타크래프트와 같은 RTS(Real-Time Strategy)에 도전하겠다고 발표했다. 하지만 RTS와 같은 게임방식의 인공지능을 머신러닝으로 구현 할 경우 복잡도가 폭발적으로 증가하기 때문에 어떠한 방식으로 구현할지 이목을 끌고 있다[7].

2.4 선행 윗놀이 인공지능 연구와의 차이점

선행 연구된 윗놀이 인공지능과의 비교를 위하여 [4]에서 구현한 인공지능과 비교해 보면, 해당 연구에서는 반복문을 이용해서 윗놀이를 이기기 위한 전략을 가정한 후 각 전략(업기, 상대편말 잡기, 지름길로 가기 등)마다 가중치를 다르게 부여하여 마지막에는 이를 더해서 가장 높은 가중치를 갖는 말이 선택되도록 인공지능을 구현 하고 있다. 즉, 상대방 말을 잡을 수 있다고 가정했을 때 가중치를 부여하고 윗놀이 판에 있는 지름길로 갈 수 있을 때 가중치를 부여하여 최종적으로는 각 말의 가중치를 더해서 비교한다. 하지만 본 논문에서는 퍼지논리 윗놀이

이 인공지능을 구현하여, 다량의 반복문의 반복을 통해서 가중치를 부여할 필요 없이 퍼지함수와 퍼지 연산을 통하여 보다 간단하게 각 말이 가지는 최적화된 수치를 계산할 수 있음을 보여준다. 따라서 본 논문은, 퍼지논리 기반 윗놀이 인공지능을 구현하여 각 상황마다 가중치를 부여하는 전통적인 사고방식에서 벗어나 각 전략 사이의 상호관계를 파악하고 조절하여 최적의 값을 산출하는 새로운 방식을 제안한다.

III. 윗놀이 인공지능 개발을 위한 배경 이론

3.1 윗놀이를 위한 인공지능 고려사항

어떠한 게임의 인공지능을 만들 때 고려해야 하는 것 중 하나는 그 게임이 완전 정보(Perfect Information)인가 불완전 정보(Non-perfect Information)인가 하는 것이다[2]. 이는 게임의 진행방식, 규칙 등에 따라 정해진다. 바둑, 체스와 같은 게임은 완전 정보 방식 그리고 윗놀이, 주사위게임등과 같은 게임은 불5완전 정보방식이다. 주로 불완전 정보 방식은 많은 경우의 수를 가지고 있기 때문에 모든 경우를 고려하는 방법은 효율적이지 않다. 그래서 본 연구에서는 이점을 극복하기 위해서 퍼지로지학을 이용하여 윗놀이 인공지능을 구현한다[4-13].

3.2 퍼지 논리

퍼지로지학은 애매모호한 상태를 컴퓨터에 표현하기 위한 방법으로 예를 들어 “얼마나 짧은가.,” “얼마나 늙었는가.”와 같은 0과1로는 표현할 수 없는 개념을 표현하기에 적합하다. 그 방법은 수학의 집합의 개념을 차용하여 사용하여 응용하여 사용하며

그래서 퍼지로직은 많은 부분에서 수학에서의 집합과 비슷하다. 그리고 퍼지로직은 $f_A(x)$ (x : 표현하고자 하는 성질의 값)이와 같이 함수처럼 표현되어지기도 하고 아래와 같이 그래프로도 표현할 수 있다. 하지만 수학적 집합의 개념에서는 0(비 포함)과 1(포함)의 관계만을 갖는 것에 반해 퍼지로직에서는 0에서 1사이의 값을 가질 수 있으며 이는 멤버십 치(Membership grade)라고 불리고 어느 정도 그 성질을 나타내는가에 관한 척도로 사용된다 [14-16].

3.3 퍼지 연산

퍼지로직에서는 수학에서의 집합 연산과 마찬가지로 퍼지연산(Fuzzy operation)이라는 개념이 존재하며 이는 수학에서의 개념과 유사점을 가지고 있다[14-16].

공집합. 퍼지집합에서도 원소의 개수가 명확하게 0일 경우 $f_A = \emptyset$ 과 같이 표현할 수 있다.

퍼지 합집합. 두 퍼지집합 f_A 와 f_B 의 합집합 f_C 은 아래와 같이 표현되며

$$f_C = \max[f_A, f_B] \quad (1)$$

수학에서의 $C = A \cup B$ 과 유사하다. 의미는 두 개의 비교성분 중 더 큰 값을 취하고 약자로는 아래와 같이 표현한다.

$$f_C = f_A \vee f_B \quad (2)$$

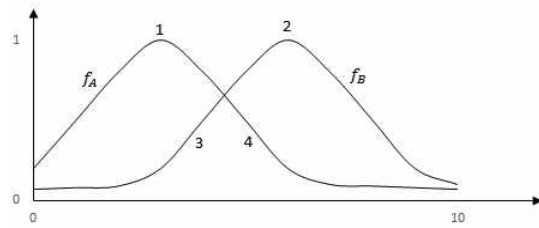
퍼지 교집합. 두 퍼지집합 f_A 와 f_B 의 교집합 f_C 은 아래와 같이 표현되며

$$f_C = \min[f_A, f_B] \quad (3)$$

이것은 수학에서의 $C = A \cap B$ 와 같고 약자로 다음과 같이 나타낼 수 있다. 의미는 두 퍼지집합 중에서 작은 값을 선택한다는 것을 의미한다.

$$f_C = f_A \wedge f_B \quad (4)$$

이 두 퍼지연산은 아래의 그림 1과 같이 그래프로 표현되어 질 수도 있다. (합집합; 1+2, 교집합; 3+4)



<그림 1> 두 퍼지함수 f_A, f_B 의 연산 그래프

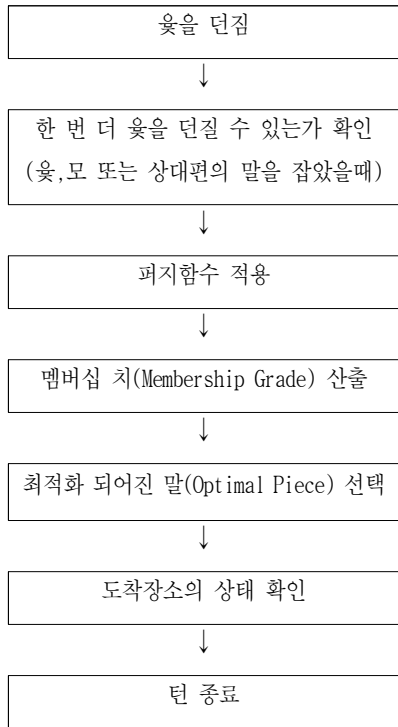
IV. 퍼지논리를 이용한 인공지능 구현방법

4.1 구현 목적

본 논문의 목적은 퍼지논리로 구현된 인공지능이 얼마만큼 인간의 선택과 유사성을 알아보기 위함이다. 또한 어느 상황에서 퍼지 논리의 선택이 한계를 갖는지 연구하기 위함이다.

4.2 게임 구성

본 논문에서 구현된 윗놀이 인공지능 및 게임구성은 아래의 <그림 2>와 같다. 우선 윗을 던져 그 값을 얻고, 한 번 더 던질 수 있는지 확인 후 이 값들과 상대편 플레이어의 말과의 관계 등의 상황을 고려한 후 퍼지로직에 의해서 가장 적합한 값을 가진 말을 선택한다.



<그림 2> 게임 구성 순서 설명

4.3 윗놀이 인공지능에 적용된 퍼지함수 제작 방법

퍼지론적으로 구현한 인공지능의 특징은 사용자의 경험이나 지식, 직관력이 인공지능에 반영되어 있다는 것이다. 이는 신경망(Neural Network)에 비해 시간과 자원을 단축시킬 수 있으며 구현은 단순하지만 강력한 성능을 가질 수 있다고 생각한다.

위의 3.1 에서 설명한 바와 같이 윗놀이는 불완전 정보 게임이고 몇몇의 독특한 규칙들과 각 플레이어 별로 4개의 말을 움직일 수 있기 때문에 다양한 전술을 펼칠 수 있다. 하지만 이는 고려해야 할 공간의 크기를 증가시키는 요인이 되고, 이는 완전정복 알고리즘과 같은 방식이 윗놀이를 위해서는 비효율적임을 의미한다. 이에 본 논문에서는 윗놀이를 이기기 위함에 있어 가장 중요한 4개의 조건과 상황을

정하고 이에 따른 퍼지함수의 제작을 통해 인공지능을 제작한다. 이 프로그램에서 제작된 퍼지함수는 4가지로 아래와 같다.

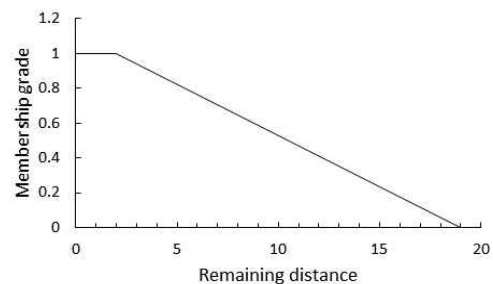
<표 1> 퍼지 함수에 관한 설명

번호	퍼지함수의 목적
1	각 말별로 현재 위치부터 골 위치까지 남은 거리에 따른 퍼지함수.
2	상대편의 말을 잡을 수 있는 확률에 따른 퍼지함수.
3	지름길로 갈 수 있는 확률에 따른 퍼지함수.
4	다른 말과 엮을 수 있는 확률에 따른 퍼지함수

이 퍼지함수들에 의해 각각의 말은 멤버십 치(Membership Grade)를 부여받으며 이 멤버십 치의 퍼지연산을 통해 최적화 된 말 A가 선택되어 진다. 이 퍼지연산 과정은 각 퍼지함수 설명 후 마지막에 제시한다.

4.3.1 남은 거리에 따른 퍼지함수

남은 거리에 따른 퍼지론적은 각각의 말부터 골 지점까지 남은거리를 계산하여 만들어진 퍼지 함수로 거리가 적게 남을수록 높은 멤버십 치(Membership Grade)를 갖는다. 이 퍼지함수는 수식 (5)과 같은 수식을 갖고 아래의 그래프<그림 3>와 같이 표현할 수 있다.



<그림 3> 남은 거리에 따른 퍼지함수의 그래프

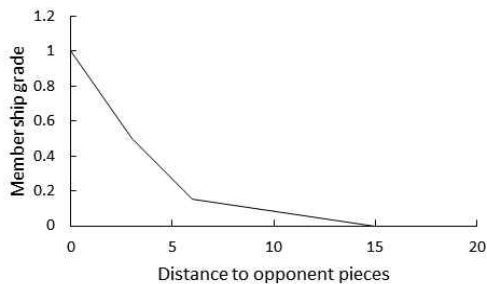
$$Rd(x) = \begin{cases} 1 & x < 2 \\ -0.1x + 1.2 & 3 \leq x \leq 10 \end{cases} \quad (5)$$

(5)와 같이 정의한 이유는 'x'는 골 지점까지의 거리를 의미하며 골 지점으로부터 3칸 이내에 위치할 경우 0.87(14/16)의 확률로 골인지점에 도착할 수 있고 이 확률은 비교적 높다고 생각되어졌기 때문에 멤버십 치중 가장 큰 값인 1을 부여하였다. 그리고 골부터 멀어질수록 멤버십 치가 완만하게 감소하도록 정의했다.

4.3.2 상대편의 말을 잡을 수 있는 확률에 따른 퍼지함수

이 퍼지함수는 상대편의 말과의 거리에 따라서 상대방 말을 잡을 확률을 추측하여 멤버십 치 (membership grade)를 부여하는 퍼지함수이다.

이 퍼지함수는 수식(6)과 같으며 <그림 4>와 같이 표현되어 질 수 있다. 이와 같이 정의된 이유는 x축의 값이(상대방 말까지의 거리) 3이하이고 다음 턴에 상대방의 말이 움직이지 않는다고 가정 할 경우 0.25(도, 걸의 확률)이상의 확률로 상대방의 말을 잡을 수 있기 때문이다. x가 4 ≤ x ≤ 5 일 경우에는 위와 같은 조건에서 한 번의 순서 이내에 잡을 수 있기 때문에 5 ≤ x 의 구간에 비해서 완만한 경사를 갖도록 했다.

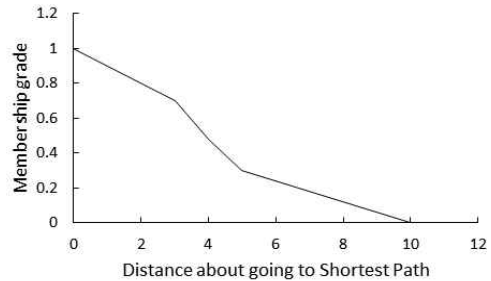


<그림 4> 상대방의 말을 잡을 수 있는 확률에 따른 퍼지함수의 그래프

$$Pc(x) = \begin{cases} -0.1x + 1.1 & x < 3 \\ -0.15x + 1.1 & 4 \leq x \leq 5 \\ -0.02x + 0.2 & 5 \leq x \leq 10 \\ -0.01x + 0.2 & 11 \leq x \end{cases} \quad (6)$$

4.3.3 지름길로 갈 수 있는 확률에 따른 퍼지 로직

이것은 윗놀이 판의 각 모서리와 가운데 있는 지름길로 움직일 수 있는 확률을 추측하는 퍼지함수이다. 이 함수는 (7)과 같이 표현하며 그래프는 <그림 5>와 같다.



<그림 5> 지름길로 갈 수 있는 확률에 따른 퍼지함수의 그래프

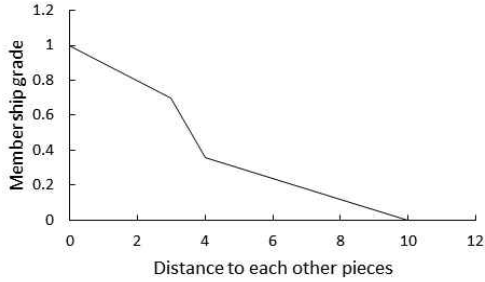
$$Ps(x) = \begin{cases} -0.1x + 1 & x < 3 \\ -0.18x + 1.2 & 4 \leq x \leq 5 \\ -0.06x + 0.6 & 6 \leq x \leq 10 \end{cases} \quad (7)$$

이 수식은 “4.3.2 상대편의 말을 잡을 수 있는 확률에 따른 퍼지함수”와 같은 방식으로 정의되었다. x축이 0일 경우 지름길에 현재의 턴에 갈 수 있는 것을 의미하고 1의 멤버십 치를 갖는다.

4.3.4 말을 얻을 수 있는 확률에 따른 퍼지 함수

이 퍼지함수는 자신의 다른 말과의 거리를 이용해서 말을 얻을 수 있는 확률을 정의한 퍼지 함수이

다. 수식(8)과 같으며 그래프는 그림6과 같이 표현할 수 있다.



<그림 6> 상대방의 말을 엮을 수 있는 확률에 따른 퍼지함수의 그래프

$$Pt(x) = \begin{cases} -0.1x + 1 & x \leq 3 \\ -0.06x + 0.6 & 4 \leq x \leq 10 \end{cases} \quad (8)$$

(8)도 위의 두 퍼지함수(4.3.2, 4.3.3)과 같은 이유로 위와 같이 정의되었다. 다른 말까지의 거리가 3 칸 이내일 경우 다음 차례에 엮을 수 있는 확률이 0.25 이상임을 고려하였다. x 축이 0일 경우 바로 엮을 수 있는 것을 의미하고 1의 멤버십 치를 갖는다.

4.3 지금까지 계산된 멤버십 치를 최종적으로 이용하는 퍼지 연산

위의 퍼지함수에 따라 각각의 말은 4개의 멤버십 치 $Rd(x), Pc(x), Ps(x), Pt(x)$ 를 갖는다. 그리고 2.2에서 제시한 퍼지로직의 합집합 연산을 이용하여 아래와 같은 퍼지연산을 하였다.

$$F = Rd(x) \vee Pc(x) \vee Ps(x) \vee Pt(x) \quad (9)$$

결론적으로 각각 말은 멤버십 치 4개와 퍼지연산 (9)를 통해서 가장 최적화 되어진 말을 선택한다. 만

약 같은 값을 갖는 경우 $Rd(x), Pc(x), Ps(x), Pt(x)$ 의 순서대로 우선순위를 갖는다. 그 이유는 남은 거리가 게임을 끝내기 위해서 가장 중요하다고 생각되어 졌기 때문이다.

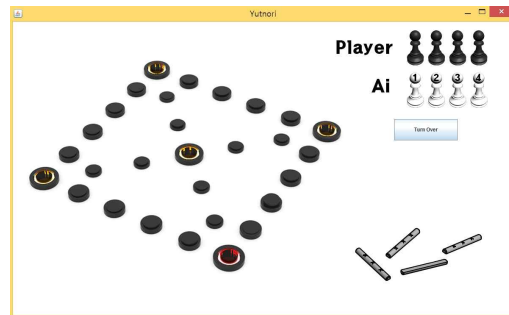
V. 실험 결과

4.1 사용자 유저 인터페이스 구현

본 논문에서는 이 인공지능의 성능평가 시 피 실험자의 편의를 제공하기 위해 GUI(Graphic User Interface)가 구현되어 졌으며 아래의 그림7과 같이 구현되어진 환경은 아래와 같다.

<표 2> GUI가 구현되어진 환경

분류	비고
OS	Windows 8 x64
Language	JAVA
Tools	Eclipse
Graphic Tools	3D Cad, Photoshop 6
CPU	Intel i5-5200u
Memory	8GB



<그림 7> Graphic User Interface

동작과정은 오른쪽 하단의 윗을 클릭하면 윗의 값을 얻을 수 있고, 플레이어가 턴을 끝낸 후 다시 윗 버튼을 클릭하면 최적 값을 가진 말이 표시되어 진다. 각각의 3D Object는 3D CAD, Photoshop을 이용해 디자인 하였다.

4.2 게임 결과 분석

본 논문의 연구 목표는 퍼지로직을 사용하여 인공지능을 구현하고 그 인공지능이 인간의 선택과 얼마만큼의 유사성을 갖는가에 관한 것이다. 그래서 이를 위해 3가지 종류의 플레이어(Human, AI, Random)를 가정했다. Human Player는 윗놀이에 대한 지식이 있는 인간 플레이어를 의미하고 AI Player는 본 논문에서 사용된 인공지능 플레이어, Random Player는 매 턴마다 임의로 말을 선택하는 플레이어를 의미한다.

4.2.1 AI Player vs Random Player

Human Player 와 Random Player 사이의 게임에서는 <표 3>와 같은 결과를 보였다. 이로 인해 얻고자 하는 것은 본 논문의 인공지능이 “기초적인 지능을 가지고 있다”는 것을 보여준다.

<표 3> 인공지능, 랜덤 플레이어의 게임결과

	vs AI	vs Random
AI player	X	20
Random Player	0	X

4.2.2 AI Player vs Human Player

<표 4> 10개의 가정된 게임 상황에 따른 설문조사

상황	말1	말2	말3	말4	Human	AI
1	0	37	X	X	2	2
2	0	37	X	X	2	2
3	14	15	8	X	2	1
4	15	22	0	X	2	2
5	0	0	37	X	3	3
6	37	0	0	X	1	2
7	0	0	X	37	4	4
8	37	X	X	0	1	1
9	18	X	X	19	4	1
10	20	X	X	17	1	4

위의 <표 4>은 윗놀이에서 생길 수 있는 상황 10가지를 임의로 선정하여 윗놀이 규칙을 알고 있는 37명의 대학생에게 “4개의 말 중 어느 말을 움직이기를 원하는가?”에 대한 설문조사이다. X가 되어있는 칸의 의미는 이미 골 지점으로 들어온 말이나 아직 출발하지 않은 말 중 상태가 겹치기 때문에 선택해도 의미가 없는 말을 의미한다.

왼쪽으로부터 2번째 열은 인간이 생각하는 최적 값이 가장 높은 말을 의미하며 4개의 말 중 가장 많이 선택되어진 말을 가장 최적화 되었다고 가정하였다.

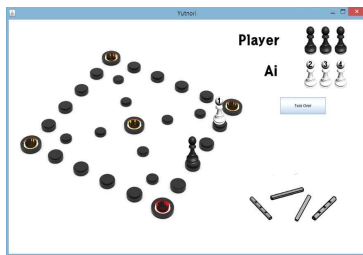
전체 결과에서 상황 1, 2, 5, 6, 7, 8은 인간의 선택이 일치하는 결과를 보여주었다. 왜냐하면 이는 이 상황에서 선택되어진 말이 다른 말들에 비해서 비교적 높은 최적 값을 가지고 있기 때문이고, 이것은 게임을 이기기 위해서는 높은 확률로 선택되어야 함을 의미하기 때문이다. 이 상황에서 본 논문의 인공지능은 6개의 가정된 상황 중 5개에서 인간의 선택과 구현한 인공지능의 선택이 일치함을 보였다.

상황 3, 4, 9, 10 은 전략, 개인의 취향에 따라 다른 최적 값을 가지는 상황이다. 이 상황 역시 다수의 사람이 선택한 말이 최적화 되어진 말이라고 가정할

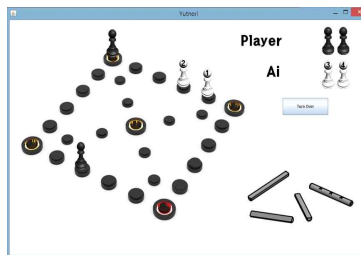
경우 4개의 선택 중 1개가 일치함을 보였다. 하지만 AI가 선택한 말이 인간이 선택하지 않은 말이 아니라 상황 3,9,10에서 봤을 때 가장 많이 선택되어진 말과 비등할 정도의 사람들이 선택한 말을 선택했다.

아래의 <그림 8>은 <표 4>에서 제시된 각 상황을

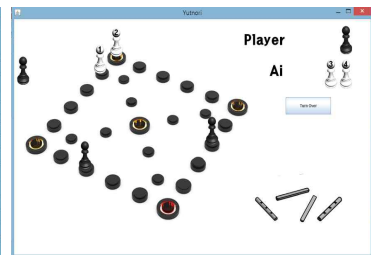
나타낸 것이고<그림 9>는 각각의 상황들의 결과를 그래프로 제시한 것이다. 위의 그래프는 인간의 선택을 의미하며 아래의 그래프는 AI의 선택을 의미한다. 각 상황에서 가장 높은 수치의 아래와 위의 색상이 같을 경우 AI가 인간과 같은 선택을 하였다고 가정하였다.



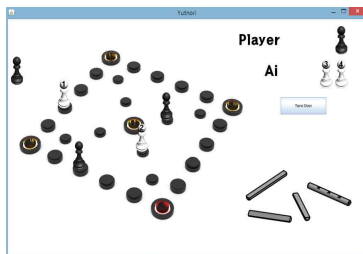
(a)상황1(웃셈:개)



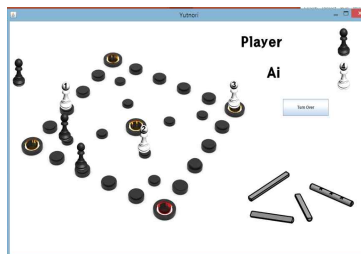
(b)상황2(웃셈:걸)



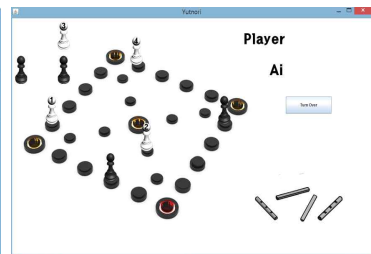
(c)상황3(웃셈:개)



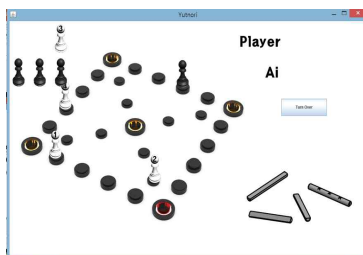
(d)상황4(웃셈:걸)



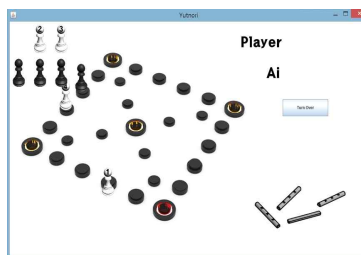
(e)상황5(웃셈:걸)



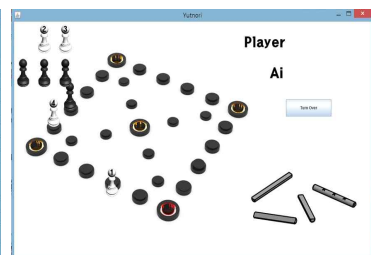
(f)상황6(웃셈:개)



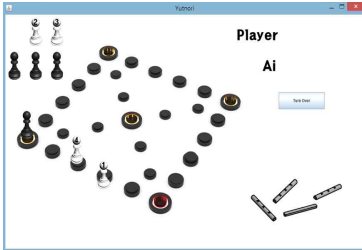
(g)상황7(웃셈:걸)



(h)상황8(웃셈:도)

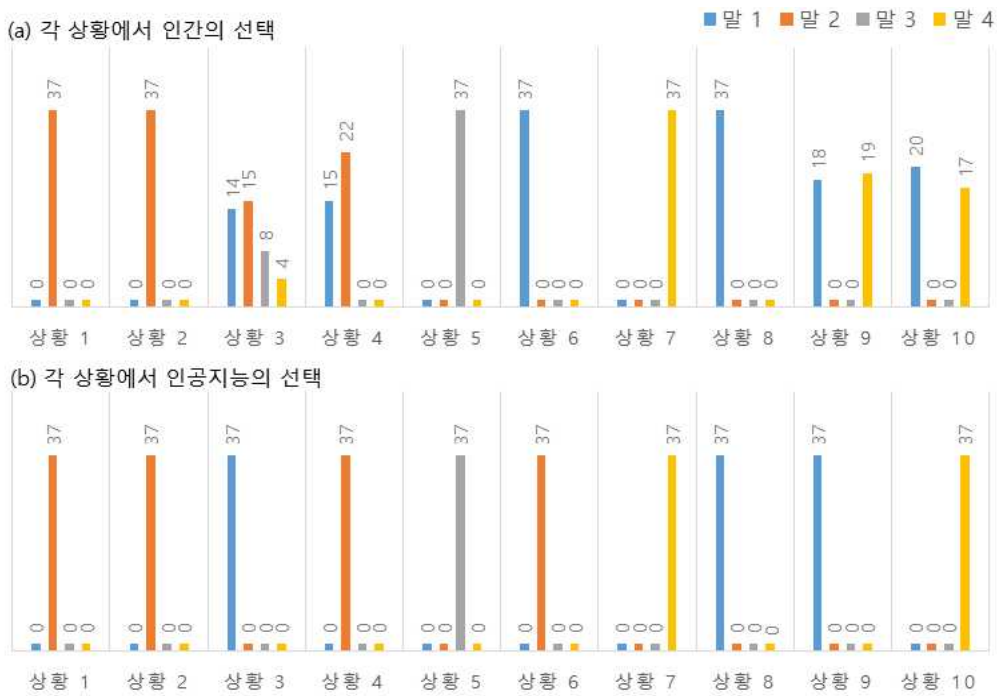


(i)상황9(웃셈:걸)



(j) 상황10(윗셈:도)

<그림 8> 게임의 성능평가에 이용된 각각의 상황설명



<그림 9> 10개의 가정된 상황에 따른 인간 플레이어 선택과 인공지능 플레이어 선택의 비교

VI. 결론

최근 알파고로 인해 인공지능에 대한 관심이 증대되는 추세이고 이에 따라 본 논문에서는 인공지능을 구현하는 한 방법인 퍼지논리를 통해서 인공지능

을 구현했다. 그 방법은 윗놀이에 대한 지식을 바탕으로 퍼지함수를 제작 한 후 이를 통해 멤버십 치를 산출한다. 그리고 이 멤버십 치를 이용하여 퍼지연산을 통해 최적 값을 가진 말을 선택하도록 구현하였다. 그리고 제작된 인공지능의 성능평가를 위해

10가지의 윗놀이에서 발생 할 수 있는 상황을 가정하고 이를 통해 인간의 선택과 얼마나 유사한지 유사성을 제시했다.

그리고 향후 본 연구를 바탕으로 퍼지논리와 딥러닝을 복합적으로 사용하는 윗놀이 인공지능을 개발할 예정에 있지만 이는 다음과 같은 문제를 가지고 있다. 알고리즘에서는 바둑의 기보라 불리는 정형화된 다량의 데이터를 이용해 딥러닝을 구현할 수 있었다.

하지만 윗놀이는 이와 같은 다량의 정형화된 데이터가 없기 때문에 본 논문의 인공지능을 바탕으로 데이터를 생성할 것이다. 하지만 여기에서 불필요한 데이터까지 성능향상에 이용되는 과적합(Overfitting) 문제나 서로 비슷한 수준의 인공지능의 결과로 만든 데이터로 인공지능을 제작 시 일정 수준 이상으로 성능이 향상되지 않는 성능정체 문제가 생길 것으로 예상된다. 이를 해결하기 위해 다양한 시도가 필요하다고 생각된다.

끝으로 이 연구를 통해 향후 윗놀이 인공지능 개발뿐만 아니라 윗놀이와 같은 불완전정보 게임의 인공지능 제작 시 방향을 제시 할 수 있는 자료가 될 수 있을 것이라고 기대된다.

참고문헌

- [1] 최예림, 김관호, "인공지능 개요 및 적용 사례," IE 매거진, 제23권, 2호, 2016, pp.23-29.
- [2] http://en.wikipedia.org/wiki/Game_theory
- [3] David Silver, Aja Huang, Chris J. Maddison et al., "Mastering the game of Go with deep neural networks and tree search," Nature, Vol. 529, Issue 7587, 2016, pp.484-489.
- [4] 오규협, 김애경, 강민숙, "윗놀이 전략 시뮬레이션," 대한산업공학회 추계학술대회 논문집, 2009, pp. 715-722.
- [5] 박현수, 김경중, "게임 인공지능 최신 연구 동향," 한국정보과학회지, 제31권, 7호, 2013, pp. 8-15.
- [6] 안일준, 박인규, "유전 알고리즘과 게임 트리를 병합한 오목 인공지능 설계 및 GPU 기반 병렬 처리 기법," 한국정보과학회지, 2010, pp. 66-75.
- [7] 이상호, 허준영, 조유근 등, "스타크래프트 인공지능의 성능 향상을 위한 불확실한 지식에 기반을 둔 적응성 있는 추론 방법," 한국컴퓨터게임학회지, 제7권, 2005, pp. 14-26.
- [8] 문성은, 장수범, 이정혁 등, "기계학습 및 딥러닝 기술동향," 한국통신학회지, 제33권, 10호, 2016, pp. 49-56.
- [9] 민병철, 김동한, 김기열 등, "퍼지제어를 이용한 바이오린 연주 연습 알고리즘 개발," 한국지능시스템학회 논문지, 제19권, 4호, 2009, pp. 550-555.
- [10] 최우경, 김성주, 전홍태, "퍼지 로직을 이용한 인간 친화적인 경로 설정에 관한 연구," 퍼지 및 지능 시스템학회 논문지, 제16권, 제5호, 2006, pp. 616-621.
- [11] 함준석, 박준형, 고일주, "비행슈팅 게임의 만족도 향상을 위한 인공지능의 개발과 적용," 디지털산업정보학회지, 제4권, 1호, 2008, pp. 55-64.
- [12] 조성래, 성행남, 안병혁, "의사결정트리와 인공지능경망 기법을 이용한 침입탐지 효율성 비교연구," 디지털산업정보학회지, 제11권, 4호, 2015, pp. 33-45.
- [13] 박종범, "인공지능 포커게임수행엔진 개발," 한국컴퓨터정보학회논문지, 제14권, 11호, 2009, pp. 41-52.
- [14] Stuart J. Russell, Peter Norvig, "Artificial Intelligence," Artificial Intelligence A Modern

Approach, 2nd ed, 2003, pp. 1-32.

[15] James C. Bezdek, Sanker K. Pal, "Fuzzy Model for Pattern Recognition," IEEE Press, New York, 1999, pp. 1-28.

[16] A. Kaufman, "Introduction to the Theory of Fuzzy Subsets," Vol. 1, Academic Press, Paris, 1973, pp. 191-264.

■ 저자소개 ■



정 성 욱
(Chung Sungwook)

2012년 3월 ~ 현재 : 창원대학교 컴퓨터공학과
조교수
2010년 10월~2012년 2월 : KT 종합기술원
중앙연구소 선임연구원
2010년 8월 : CISE dept. Univ. of Florida, USA,
(Ph.D)
2005년 5월 : CISE dept. Univ. of Florida, USA,
(MS)

관심분야 : 분산멀티미디어시스템, 홈 네트워크
E-mail : swchung@changwon.ac.kr



김 기 년
(Kim Kinyun)

20010년 3월~현재 : 창원대학교 컴퓨터공학과
학사

관심분야 : 게임 프로그래밍, 인공지능
E-mail : 71year@changwon.ac.kr

논문접수일 : 2016년 12월 08일
수 정 일 : 2016년 12월 22일
게재확정일 : 2016년 12월 30일