

# 퍼지이론을 이용한 슈팅게임 난이도 조절

박창훈<sup>†</sup>, 서진석<sup>\*\*</sup>

## Difficulty Control of a Scrolling-Shooter Game Using Fuzzy Reasoning

Chang Hoon Park<sup>†</sup>, Jinseok Seo<sup>\*\*</sup>

### ABSTRACT

One of the important factors in game design is difficulty adjustment. An appropriate level of difficulty makes users have a sense of challenge and interest. However, the adjustment of difficulty takes a lot of time and effort, because of its ambiguity. To solve the problem, we propose a difficulty control method using a fuzzy theory. In this paper, a simple demonstration is exemplified to verify the effectiveness of our method. Experimental results show that the difficulty of the game changes according to the user's skill.

**Key words:** Scroll-Shooter Game, Game Difficulty Control, Fuzzy Reasoning

### 1. 서론

비행 슈팅 게임을 디자인하는 게임 디자이너는 난이도 조절을 위해 아이템의 생성 확률을 조정하거나 사용자가 사용할 수 있는 비행기의 개수를 제한한다. 또한, 게임 디자이너는 게임의 난이도를 여러 단계로 나누어 난이도가 서서히 상승되도록 한다. 단계가 높아질수록 더욱 복잡한 패턴의 움직임을 갖는 강력한 비행기를 등장시킨다. 하지만 높은 단계에서 어느 정도의 복잡한 움직임 패턴을 사용할 지, 어느 정도 강하게 조절할 지 정확하게 산출하는 것은 매우 어려운 일이다[1]. 하물며 '복잡한'이나 '강한' 같은 개념은 산술적으로 정의하기가 모호하다. 디자이너는 모호한 부분을 정의하고 난이도 조절 값을 산출하기 위해 많은 시간을 소모한다. 이는 디자이너가 적당한 난이도라고 생각될 때까지 수많은 테스트를 반복하기 때문이다. 더불어 다양한 사용자층을 만족시키기

위해서는 더욱 많은 테스트 시간이 필요하다.

위와 같은 어려움을 줄이고자, 본 논문에서는 퍼지 이론을 이용한 비행 슈팅 게임의 난이도 조절 시스템을 제안한다. 또한, 이를 바탕으로 제작된 데모를 통해 시스템을 평가하여 실제 게임에서 적용될 수 있음을 보여준다.

### 2. 관련 연구

게임 내부에서 동적으로 콘텐츠를 생산하는 연구는 이전부터 활발하게 진행되어 왔다. 이러한 분야 중 하나가 PCG(Procedural Content Generation)이다. 강신진은 PCG의 다양한 연구 주제와 응용 사례들을 소개하고 있다[2]. Thureau는 일반적인 NPC가 가진 한계를 극복하기 위한 새로운 기법을 제안하였다. 새로운 기법이 사용된 NPC는 사용자의 행동패턴을 습득하고 새로운 행동 패턴을 생성한다. NPC의

※ Corresponding Author : Jinseok Seo, Address: 176 Eomgwangno, Busan Jin-gu, Busan 47340, Korea, TEL : +82-51-890-2712, FAX : +82-505-182-6907, E-mail : jsseo@deu.ac.kr  
Receipt date : Jul. 17, 2017, Approval date : Aug. 3, 2017

<sup>†</sup> Dept. of Digital Media, Dong-Eui University  
(E-mail : katze9027@gmail.com)

<sup>\*\*</sup> College of ICT Engineering, Dong-Eui University

※ This work was supported by Dong-eui University Grant.(201703040001)

기계 학습에는 Self-Organized map과 Bayesian Network가 사용되었다[3,4].

Nicholas는 유전자 알고리즘을 전략 게임에 적용하여 동적으로 난이도가 유지되도록 하였다[5]. 이 연구에서는 1인칭 슈팅 게임의 인공지능에 필요한 파라미터를 동적으로 생성한다. 업상원은 유전자 알고리즘을 테트리스 게임의 난이도 조절에 사용하였다[6]. 이 연구는 사용자의 숙련도에 따라 테트리스 블록의 생성 패턴을 변화시킨다(Fig. 1 참조). 문성원 등은 제한된 자원을 사용하는 온라인 게임에 적합한 지능적 AI 제작 기법을 연구하였다[7]. 이 연구는 퍼지 확장 기법을 사용하여 시스템에 발생하는 부하가 적도록 하였으며, 좀 더 인간에 가까운 AI 구현이 가능하도록 하였다(Fig. 2 참조).

양정도 등은 2D 액션 게임의 NPC가 사용자의 다양한 행동에 맞추어 반응할 수 있도록 하였다. NPC는 동적 FSM(Finite State Machine)으로 구현되었으며, 사용자의 행동정보를 기반으로 동작한다[8]. 장새롭 등은 사용자의 게임 행위를 분석하고, 이를 기반으로 게임 환경을 생성할 수 있게 하였다. 게임

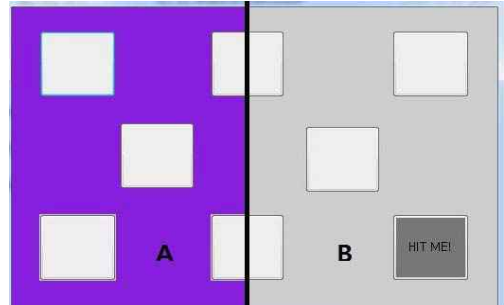


Fig. 3. Game using artificial emotion.

환경 생성 시스템은 사용자가 이동한 좌표 데이터를 수치화하며, 이를 기반으로 맵의 형태와 객체들의 위치를 조절한다[9]. 또한, 이들은 퍼지이론을 이용하여 사용자가 NPC의 행동 패턴을 예측하기 어렵게 하는 연구도 진행하였다[10]. 박준형 등은 인공감정을 게임에 적용하여 난이도 조절이 가능하도록 하였다. 사용자의 행동은 인공 감정 에이전트에 감정 자극을 주며, 변화된 에이전트의 감정은 게임의 난이도를 조절하는데 사용된다[11].

### 3. 난이도 조절 퍼지 시스템

#### 3.1 제안 시스템 개요

본 연구에서는 슈팅 게임의 난이도 조절을 PCG의 관점에서 접근하였다. 제안하는 시스템은 퍼지 이론을 이용하여 구현되었다. 퍼지 이론은 이동패턴의 '복잡함'이나 적 비행기의 '강함'과 같은 이분법적으로 표현하기 힘든 애매한 표현을 처리하기에 적합하다.

제안된 시스템은 맘다니의 퍼지 추론 단계를 따른다. 먼저 사용자의 실력을 수치화하기 위해 플레이 경과 시간, 명중률, 적 비행기 파괴 시간 간격, 공격력 향상 아이템 사용 비율과 같이 네 가지 항목에서 데이터를 수집한다. 수집된 값은 퍼지화 단계를 거치면서 각 항목의 퍼지 집합들에 대한 소속도로 계산된다. 시스템은 평가 단계에서 계산된 소속도를 이용하여 규칙을 평가하고 통합 단계에서 전항의 평가 결과들을 후항의 퍼지 집합으로 축약한다. 여기서 생성된 신뢰도는 역퍼지화를 거친 후 적 비행기의 능력을 변화시키는 값으로 사용된다. 변화되는 적 비행기의 능력은 이동속도, 생산 딜레이, 그리고 생명력이다. 측정된 사용자의 실력에 따라 적 비행기의 능력을 변화시킴으로써 난이도를 조절한다.

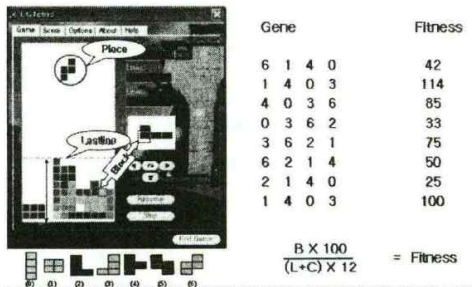


Fig. 1. Tetris game using a genetic algorithm.



Fig. 2. Online game using a Fuzzy extension principle.

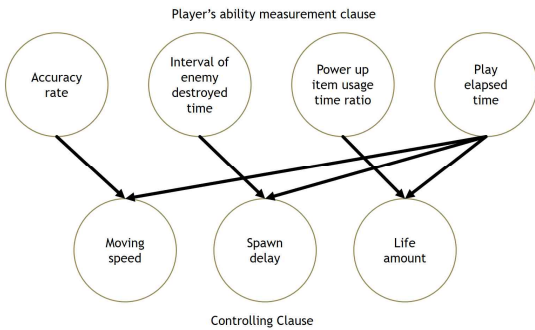


Fig. 4. Correlation between measurement clause and controlling clause.

### 3.2 사용자의 실력 측정

시스템은 네 가지 항목을 측정함으로써 사용자의 실력을 수치화한다. 첫 번째 측정 항목은 플레이 경과 시간이다. 플레이 시간이 증가할수록 난이도는 상승되도록 할 필요가 있다. 두 번째, 명중률은 사용자

비행기에서 생성된 발사체가 목표물에 명중하는 비율이며, 모든 슈팅 게임에서 사용하는 판정 기준이다. 세 번째, 적 비행기 파괴 간격은 사용자가 느끼는 긴장감을 측정하기 위한 것이다. 간격이 짧다는 것은 적 비행기의 수가 사용자에게 긴장감을 주기에 부족하다는 것과 같다. 따라서 시스템은 적 비행기를 더 빠르게 생성할 필요가 있다. 마지막으로, 공격력 향상 아이템 사용 비율은 플레이 환경이 적 비행기를 파괴하기에 유리한 상황인지 아닌지를 판별하는 데 사용된다. Fig. 4에 사용자의 실력을 수치화하기 위한 측정 항목과 조절되는 적 비행기의 항목 간 상관 관계를 나타내었다.

### 3.3 퍼지화

퍼지화는 퍼지 집합체의 모든 퍼지 집합에 대하여 소속도를 구하는 것이다. Fig. 5와 Fig. 6에 본 논문의 퍼지화 과정에서 사용되는 퍼지 집합체와 소속도 함



Fig. 5. Fuzzy sets of measurement item.

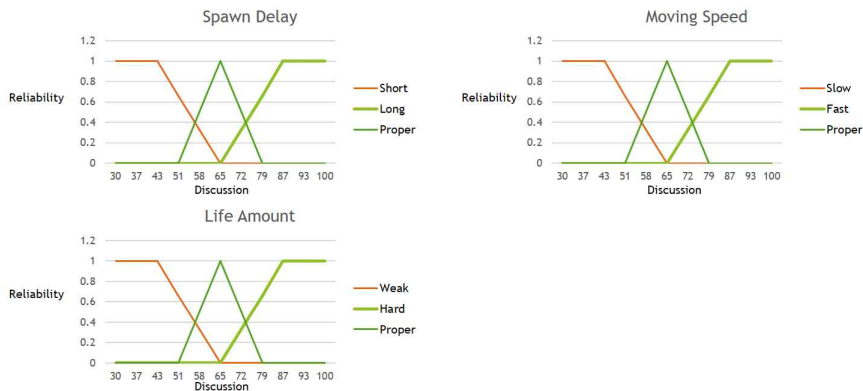


Fig. 6. Fuzzy sets of enemy's ability.

Table 1. Discussion and unit of Fuzzy sets

	Accuracy rate	Power up item usage time ratio	Interval of enemy destroyed time	Play elapsed time
Unit	Percent(%)		Sec	
Discussion range	0 - 100		0.0 - 6.0	0 - 600

Table 2. Discussion and unit of Fuzzy sets

	Moving speed	Spawn delay	Life amount
Unit	Percent(%)		
Discussion range	30 - 100		

수를 나타내었다. 또한, 퍼지화가 이루어지는 논의범위를 Table 1와 Table 2에 나타내었다.

명중률 55%는 Bad 퍼지 집합에서는 0, Good 퍼지 집합에서는 0.75, Awesome 퍼지 집합에서는 0.165의 소속도로 계산된다. 시스템은 소속도 계산 과정에서 논의 범위의 최댓값을 넘는 수치가 입력될 경우, 소속도를 1로 클리핑 하여 사용한다.

### 3.4 규칙 평가

규칙 평가는 적 비행기가 생산된 직후 진행된다. 퍼지 규칙은 퍼지 집합체 사이의 관계를 나타낸 것으로, "IF Accuracy rate is Good AND Play elapsed time is Long THEN 'Moving speed is Fast.'"의 형태를 가진다. 각 퍼지 집합체 간의 퍼지 규칙은 Fig. 7에 나타나 있다.

전항의 소속도는 규칙 평가 단계를 통해 후항의

신뢰도로 바뀐다. 만약 규칙 평가 단계에서 전항이 두 개 이상이면 퍼지 연산을 이용하여 하나로 축약한다. 일반적인 퍼지 시스템에서는 두 개 이상의 평가 결과를 하나로 만들기 위해 합집합인 OR 연산을 사용한다. 하지만 본 논문의 퍼지 시스템은 두 개의 전항을 모두 만족하는 결과를 사용하므로, 교집합인 AND 연산을 사용한다.

게임이 시작한지 500초가 경과되고 명중률이 55%일 때, 퍼지 규칙 중 하나인 "IF Accuracy rate is Good AND Play elapsed time is Long THEN Moving speed is Fast."를 평가하면 Table 3과 같다. 전항인 명중률의 Good 소속도는 0.75, 플레이 경과 시간의 Long 소속도는 0.4이다. AND 연산은 이 중 작은 값을 가져오므로 후항인 이동속도 Fast의 신뢰도는 0.66이 된다.

### 3.5 결과 도출

모든 퍼지 규칙의 평가가 끝나면 후항에 속한 퍼지 집합의 신뢰도를 최종 결정한다. 이때 퍼지 규칙 사이에서 후항이 중복되면 경계 합 혹은 OR 연산을 사용하여 그 값을 통합한다. 본 연구에서는 합집합인 OR 연산을 사용한다. Slow의 경우 0, 0.4, 0.75와 같이 후항의 신뢰도가 세 번 중복된다. 이 중 가장 큰 값을 가져오므로 퍼지 집합 Slow의 신뢰도는 0.75이다. Table 4는 Table 3의 중복된 신뢰도를 정리하여 나타낸 것이다.

Table 4와 같이 신뢰도가 결정된 퍼지 집합체는 역퍼지화가 진행된다. 역퍼지화 방법으로, 본 연구에서는 무게중심법보다 계산이 간단하고, 그에 근접한 값의 도출이 가능한 최댓값들의 평균법을 사용한다. 최댓값들의 평균법으로 크리스프 출력을 구하는 공

Fuzzy rule for calculating moving speed

		Play elapsed time		
		Short	Proper	Long
Accuracy rate	Bad	Slow	Slow	Proper
	Good	Slow	Proper	Fast
	Awesome	Proper	Fast	Fast

Fuzzy rule for calculating spawn delay

		Play elapsed time		
		Short	Proper	Long
Interval of destroy enemyed time	Short	Proper	Short	Short
	Proper	Long	Proper	Short
	Fast	Long	Long	Proper

Fuzzy rule for calculating life amount

		Play elapsed time		
		Short	Proper	Long
Power up item usage time ratio	Short	Weak	Weak	Proper
	Proper	Weak	Proper	Hard
	Long	Proper	Hard	Hard

Fig. 7. Fuzzy rule's of system.

Table 3. Example of fuzzy rule evaluation

		Play elapsed time		
		Short(0.0)	Proper(0.4)	Long(0.4)
Accuracy rate	Bad(0)	Slow(0)	Slow(0.4)	Proper(0.4)
	Good(0.75)	Slow(0.75)	Proper(0.4)	Fast(0.4)
	Awesome(0.165)	Proper(0.165)	Fast(0.165)	Fast(0.165)

Table 4. Combined reliabilities of moving speed

Slow	Proper	Fast
0.75	0.4	0.4

Table 5. Representative value of Fuzzysets

Fuzzy sets			
Spawn delay	Moving speed	Life amount	Representative value
Short	Slow	Weak	36.5
Proper	Proper	Proper	65
Long	Fast	Hard	93.5

식은 수식 (1)에 나타내었으며, 역퍼지화에 사용되는 퍼지집합체의 대푯값은 Table 5에 나타내었다.

$$Crisp Value = \frac{\sum(Representative Value \times Reliability)}{\sum Reliability} \quad (1)$$

Table 4의 신뢰도와 Table 5의 대푯값을 이용하여 수식 (1)에 대입하면 이동 속도의 크리스프 출력을 얻을 수 있다. 계산된 결과 값은 단위가 백분율이기 때문에 각 능력치의 최댓값을 곱해야 한다. 생성 딜레이는 2, 이동 속도는 7, 생명력은 5가 최댓값이다. 그러므로 현재 예에서 결정된 적 비행기의 이동 속도는 3.04가 된다. 결정된 값은 다음 적 비행기 생성 시에 적용된다.

#### 4. 실험 방법

시스템의 검증을 위해 간단한 데모 게임을 구현하였다. 실험 조건은 다음과 같다. 사용자가 자연스럽게 플레이할 수 있도록 난이도 조절 조건은 알려주지 않는다. 데모 게임은 충분한 실험 데이터를 얻기 위해 사용자의 적 비행기가 파괴되더라도 게임이 끝나지 않도록 했다. 또한, 사용자의 동기부여를 위해 적 비행기 파괴 횟수와 사용자 비행기 파괴 횟수를 확인할 수 있도록 했다.

실험 데이터 수집은 2초 마다 자동으로 시행되며,

데이터는 R 분석 프로그램을 사용하기 위해 CSV (Comma-separated values) 파일 포맷으로 저장된다. 저장되는 실험 데이터는 기록 당시의 시간, 사용자의 실력 측정 결과, 적 비행기의 조절항목이다.

#### 5. 실험 결과 및 고찰

##### 5.1 이동 속도 조절 실험 결과

적 비행기의 이동 속도는 사용자의 실력 상승 혹은 플레이 경과 시간의 상승에 따라 증가하도록 설정되어 있다 (Fig. 8). (a)의 두 그래프는 명중률과 이동 속도의 관계를 나타낸 그래프로 명중률이 상승하면 이동 속도가 증가한다. 플레이 경과 시간과의 결과는 (b) 그래프를 보면 알 수 있다. 선형회귀분석 결과, 두 그래프 모두 상승하는 추세가 보였다.

마지막 (c) 그래프는 전체적인 이동속도의 변화를 보여준다. 플레이 경과 시간이 증가하더라도 명중률이 하락하면 이동속도가 감소한다. 또 반대로 게임을 시작한 지 얼마 되지 않았더라도, 명중률이 상승하면 이동속도가 상대적으로 높아졌다.

##### 5.2 생산 딜레이 조절 실험 결과

적 비행기의 생산 딜레이 조절 결과는 Fig. 9에 나타내었다. (a) 그래프는 생산 딜레이와 플레이 경과 시간 그래프, (b) 그래프는 적 비행기 파괴 간격

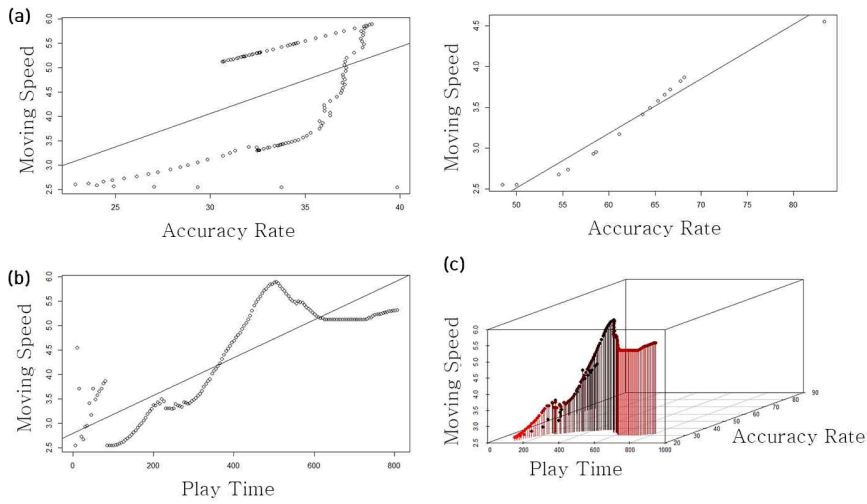


Fig. 8. Graph of moving speed control result, (a) Moving speed-Accuracy rate graph, (b) Moving speed-Play time graph, and (c) Moving speed-Play time-Accuracy rate graph.

시간과 생산 딜레이 그래프, (c) 그래프는 생산 딜레이, 플레이 경과 시간, 적 비행기 파괴 간격 시간을 축으로 하는 3차원 그래프이다. 생산 딜레이와 플레이 경과 시간은 반비례 관계이다. Fig. 9의 (a) 그래프를 보면 시간이 증가할수록 적 비행기의 생산 딜레이가 줄어드는 것을 알 수 있다. 선형회귀분석 결과, 하락하는 추세를 보인다.

생산 딜레이와 적 비행기 파괴 간격은 비례관계이다. 이는 (b) 그래프를 보면 알 수 있다. 적 비행기의 파괴 시간이 짧을수록 생산 딜레이가 줄어든다. 해당 그래프의 경우 그래프 상의 수치 분포가 매우 넓은

편이나 선형회귀분석 추세가 상승하는 것으로 보아 비례 관계가 유지되고 있음을 알 수 있다. (c) 그래프를 통해 전체적인 생산 딜레이의 변화를 확인 할 수 있다.

### 5.3 생명력 조절 결과

적 비행기의 생명력 조절 결과는 Fig. 10에 나타나 있다. 생명력과 플레이 경과 시간은 비례관계이다. Fig. 10의 (a) 그래프를 보면 플레이 시간이 지날수록 생명력이 향상되는 것을 알 수 있다. 공격력 향상 아이템 사용 비율과 생명력 또한 비례관계이다 이는

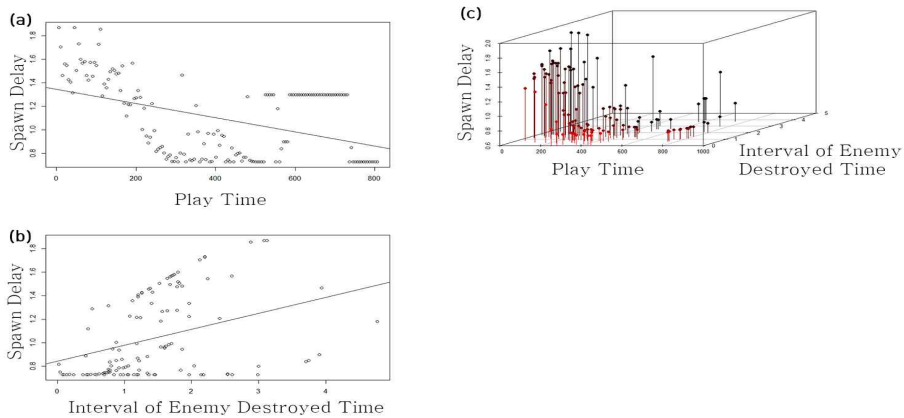


Fig. 9. Graph of spawn delay control result, (a) Spawn delay-Play time graph, (b) Spawn delay-Interval of enemy destroyed time graph, and (c) Spawn delay-Play time-Interval of enemy destroyed time graph.

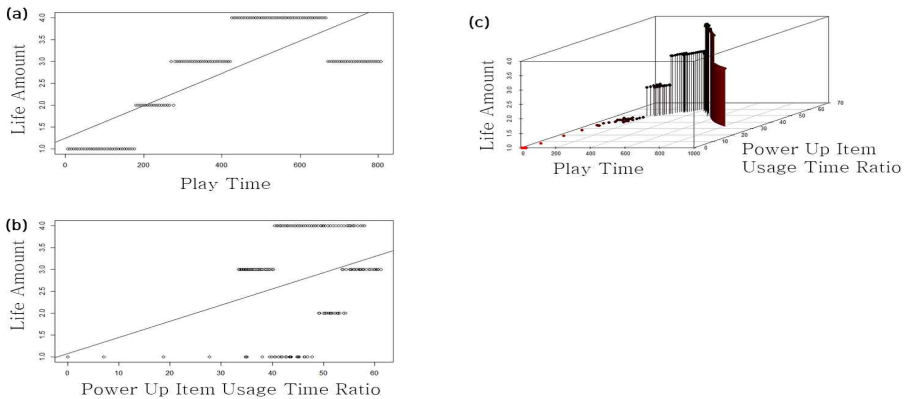


Fig. 10. Graph of life amount control result, (a) Life amount-Play time graph, (b) Life amount-Power up item usage time ratio graph, and (c) Life amount-Play time-Power up item usage time ratio.

(b) 그래프를 통해 확인 가능하며, 비율이 높을수록 생명력이 향상되는 것을 확인할 수 있다. 본 실험의 결과는 이전의 실험과 마찬가지로 분석 추세선이 상승세이며, 수치 조절이 성공적으로 진행됨을 알 수 있다. (c) 그래프는 전체적인 변화를 보여준다.

6. 결 론

본 논문에서는 사용자의 실력을 측정하고, 이를 기반으로 사용자에게 맞는 난이도 생성 시스템을 제안하였다. 제작된 데모게임을 통해 실험한 결과, 사용자의 실력이 변화하면 적 비행기의 능력 또한 변화하는 것을 확인하였다. 이는 사용자의 실력이 향상됨에 따라 강한 적 비행기가 나타나 난이도가 조절된다는 것을 의미한다. 제안된 시스템은 기존의 난이도 조절 방법보다 시간과 비용이 절감된다. 이는 테스트 과정이 줄어 실시간으로 사용자의 실력에 따른 난이도 조절이 가능하기 때문이다. 실력이 좋지 않은 사용자에게는 게임에 적응할 수 있는 시간을 제공하여 실력을 향상시킬 수 있으며, 실력이 좋은 사용자에게는 더 높은 난이도를 맞춤형으로 제공 가능하다. 본 시스템을 사용하여 게임 난이도를 조절한다면 게임의 진입장벽이 낮아져 많은 사용자가 유입될 것이며, 적절한 긴장을 제공하여 사용자의 수가 줄지 않게 할 수 있을 것으로 기대된다.

REFERENCE

[1] H. Hyun and T. Kim, "Fuzzy Based Selection

Technique for Character Action in Game Balancing," *Journal of The Korea Society of Computer and Information*, Vol. 13, No. 3, pp. 81-88, 2008.  
 [2] S. Kang, "Procedural Contents Generation Algorithm in Online Game," *Journal of Korea Multimedia Society*, Vol. 17 No. 1, pp. 1-5, 2013.  
 [3] C. Thureau and C. Bauckhage "Learning human-like Movement Behavior for Computer," *Proceedings of the 8th International Conference on Simulation of Adaptive Behavior*, pp. 315-323, 2004.  
 [4] C. Thureau, T. Paczian, and C. Bauckhage, "Is Bayesian Imitation Learning The Route to Believable Gamebot?," *Journal of Intelligent Systems Technologies and Applications*, Vol. 2 pp. 284-295, 2007.  
 [5] C. Nicholas, S.J. Louis, and C. Miles, "Using a Genetic Algorithm to Tune First-Person Shooter Bots," *Proceeding of the 8th International Congresson Evolutionary Computation*, pp. 139-145, 2004.  
 [6] S. Um, T. Kim, and J. Choi, "Player Dependent Difficulty Control Algorithm Using Genetic Algorithm," *Journal of The Korean Society for Computer Game*, Vol. 1, No. 1, pp. 7-15, 2002.  
 [7] S. Mun and H. Cho, "Intelligent AI Technique

Adaptive for Online Game Using Fuzzy Extension Principle,” *Journal of Korea Game Society*, Vol. 8, No. 3, pp. 77-85, 2008.

[8] J. Yang, K. Cho, and K. Um, “A Dynamic Utilization method of FSM for Adaptive NPC Generation,” *Journal of Korea Multimedia Society*, Vol. 11, No. 9, pp. 1258-1266, 2008.

[9] S. Jang, J. Lee, Y. Choi, M. Choo, and T. Yoon, “Method of Dynamic Game Map Generation Using Player’s Game Behavioral Data,” *Proceeding of Korea Multimedia Society*, pp. 615-616, 2009.

[10] J. Lee, S. Jang, Y. Choi, M. Choo, and T. Yoon, “Cases of XNA-based Game Development Using Fuzzy Theory : The Ring of Hitler,” *Proceeding of Korea Multimedia Society*, pp. 619-621, 2009.

[11] J. Park, H. Hyun, J. Yeo, and I. Ko, “Study of Game Level Design Controlled by Artificial Emotion,” *Proceedings of the Korean Society of Computer Information Conference*, Vol. 19, No. 1, pp. 181-183, 2011.



박 창 훈

2018년 3월~현재 동의대학교 디지털미디어공학과 석사과정  
2018년 2월 동의대학교 게임공학과 학사



서 진 석

2005년 9월~현재 동의대학교 게임애니메이션공학전공 교수  
2005년 2월~2005년 8월 포스텍 박사후연구원  
2005년 2월 포스텍 컴퓨터공학과 박사  
2000년 2월 포스텍 컴퓨터공학과 석사  
1998년 2월 건국대학교 전자계산학과 학사