

# 대전형 액션 게임에서 에너지 점수를 도입한 지능 캐릭터

이 면 섭<sup>†</sup> · 조 병 현<sup>\*\*</sup> · 정 성 훈<sup>\*\*\*</sup> · 성 영 략<sup>\*\*\*\*</sup> · 오 하 령<sup>\*\*\*\*\*</sup>

## 요 약

본 논문에서는 대전형 액션 게임을 위한 지능 캐릭터 구현에 있어서 기존 연구보다 실제적인 구현을 위하여 에너지 점수를 적용한 지능 캐릭터를 제안한다. 제안한 지능 캐릭터는 기존연구에서 사용한 상대 캐릭터의 현재 행동, 행동의 단계, 유효거리, 과거 행동 뿐 만 아니라 자신의 에너지 양을 고려하여 행동을 결정하게 된다. 에너지는 현재 온라인 게임 등에서 많이 사용하는 것으로 HP (Health Point)와 MP(Mana Point) 두 가지를 사용하였다. 에너지 점수를 적용한 지능캐릭터가 학습하는지 그리고 에너지점수를 적용했을 때 어떻게 행동하는지를 알아보기 위한 여러 가지 실험을 하였다. 실험 결과 에너지가 양이 많은 경우에는 상대방에게 피해를 크게 줄 수 있는 행동으로 대응하고 그렇지 않은 경우에는 에너지를 재충전하기 위한 행동들을 하는 것을 관찰하였다. 이런 결과로 볼 때 본 논문에서 제안한 에너지를 적용한 지능캐릭터가 잘 동작 하며 에너지를 고려하여 효과적인 반응을 보임을 알 수 있다.

키워드 : 인공지능, 대전액션게임, Health Point, Mana Point

## Intelligent Characters for Fighting Action Games applied Energy Points

Lee Myun-Sub<sup>†</sup> · Cho Byeong-heon<sup>\*\*</sup> · Jung Sung-hoon<sup>\*\*\*</sup> · Seong Yeong-rak<sup>\*\*\*\*</sup> · Oh ha-ryoung<sup>\*\*\*\*\*</sup>

## ABSTRACT

This paper proposes intelligent characters for fighting action games to which energy points are applied for more realistic implementation than those of previous researches. The intelligent characters decide their actions in consideration of their energy level as well as a current action, the step of the action, the distance, and past actions of opponent characters that were used in existing intelligent ones. We used two types of energy, HP(Health Point) and MP(Mana Point) that were frequently employed in recent on-line games. We experimented with proposed intelligent characters to investigate whether the intelligent characters learn proper actions and cope with opponent characters in consideration of their energy levels. Experimental results showed that the intelligent characters reacted with the best actions to obtain high score if their energy is sufficient, Otherwise, they did the actions to(that?) recharge their energy. From this observation, we could conclude that the proposed intelligent characters worked well and did effective actions in consideration of the their energy.

Key Words : Intelligent Character, Fighting Action Game, Health Point, Mana Point

### 1. 서 론

초고속 인터넷의 확산으로 온라인이나 네트워크 상에서 여러 명의 사용자가 동시에 접속하여 즐길 수 있는 다양한 장르의 게임이 개발되고 있다. 이 개발과정에서 게임 개발자나 제작자들은 인공지능의 중요성을 인식하고 있으며 이로 인하여 게임에서 인공지능은 중요한 비중을 차지하고 있다. 게임에 인공지능을 적용한 것으로는 Black & White, The Sims, Half Life등이 있다[1, 2, 3]. 이러한 방법들에서는

인공지능이 게임 플레이어의 지능적인 행동에 직접 관여하지는 않으며 게임을 실감나게 하기 위해 혹은 단순한 반복을 회피하기 위해 사용되고 있다. 또한 모든 상황을 미리 설정해 놓고 그 규칙대로 코딩이 이루어지므로 한번 코딩이 이루어지면 게임 중간에 규칙을 바꿀 수가 없다는 단점이 있다. 이와 같이 게임에 인공지능을 도입하려는 시도가 활발하게 이루어지고 있으나 아직 캐릭터의 행동에 대해서 인공지능을 적용하는 예는 매우 드물다.

우리는 대전형 액션 게임에서 지능 캐릭터를 구현하기 위한 연구를 수행하여 왔다[4, 5, 6]. [4]에서는 기초연구로 캐릭터의 모든 행동이 하나의 클럭으로 종료되는 게임을 대상으로 유전자 알고리즘을 이용하여 게임 규칙을 학습하고 스스로 적용해 나가는 방법을 연구하였다. [5]에서는 [4]에서 보다 실질적인 지능 캐릭터를 구현하기 위하여 캐릭터의 행동이 한 클

<sup>†</sup> 정 회 원 : 인천전문대학 컴퓨터 정보과 전임강사, 조교수, 부교수

<sup>\*\*</sup> 정 회 원 : 한국전자통신연구원 디지털콘텐츠연구단 연구원

<sup>\*\*\*</sup> 정 회 원 : 한성대학교 정보통신공학과 부교수

<sup>\*\*\*\*</sup> 정 회 원 : 국민대학교 전자정보통신공학부 부교수

<sup>\*\*\*\*\*</sup> 정 회 원 : 국민대학교 전자정보통신공학부 부교수

논문접수 : 2006년 3월 9일, 심사완료 : 2006년 6월 20일

력으로 종료되는 것이 아니라 행동의 종류에 따라 여러 개의 클럭으로 구성되는 경우를 연구하였다. 그러나 [5]에서는 상대 캐릭터의 현재 행동만을 고려하였기 때문에 상대 캐릭터가 행동 패턴을 보일 경우에 이에 대한 대응 방법을 제시하지 못하였다. 이를 극복하기 위하여 [6]에서는 상대 캐릭터가 이동 행동을 포함하여 행동 패턴을 보이는 경우에 지능 캐릭터의 효율적인 대응 방법을 제시하였다. 그러나, 위의 모든 연구에서 현재 대부분의 대전용 액션게임이나 온라인 게임에 도입된 에너지 개념은 적용하지 못하였다. 에너지 개념이 없다는 것은 에너지 사용에 제한이 없다는 뜻으로 다른 말로 무한대의 에너지를 갖은 것으로 볼 수 있다. 그러므로 상대 공격에 대하여 현재 거리에서 소요시간을 고려하여 가장 많은 점수를 획득할 수 있는 공격을 결정하기만 하면 되었다.

본 연구에서는 보다 실제적인 지능 캐릭터 구현을 위하여 대전 액션게임의 지능 캐릭터에 에너지 개념을 적용하였다. 적용한 에너지는 현재 온라인 게임에 많이 도입된 HP(Health Point)와 MP(Mana Point) 두 종류이다. HP는 상대의 공격을 받았을 경우 공격의 종류에 따라서 감소하는 에너지로 게임 종료 후 HP가 많이 남은 캐릭터가 승자가 된다. 이때 상대방의 HP가 0이면 KO승이 된다. MP는 특수기술을 사용하기 위한 에너지로 특수기술을 사용한 캐릭터는 해당 에너지만큼 MP가 감소하고 특수공격을 받은 캐릭터는 해당 에너지만큼 HP가 감소한다. 또한 대부분의 온라인 게임처럼 일정시간 동안 상대방의 공격을 피하거나 막은 경우(즉 HP 에너지를 잃지 않은 경우) HP와 MP의 에너지가 회복되는 기능을 추가하였다. 그러므로 효과적으로 학습한 지능 캐릭터는 HP와 MP가 많은 경우 해당 거리에서 상대방의 행동에 대하여 최대한의 점수를 획득하는 쪽으로 행동을 결정하며 에너지가 충분하지 않은 경우 상대방의 공격을 적절히 회피 하면서 에너지를 회복하는 행동을 선택할 것이다.

우리는 여러 가지 실험을 통하여 본 논문에서 제안한 지능캐릭터가 위와 같이 효과적인 행동을 학습하는지 알아보았다. 실험은 제안한 지능캐릭터가 제대로 학습하는지, 승률이 높게 학습된 캐릭터가 일반적으로 우수한지, 학습에 사용되지 않은 상대방과의 대결에서 얼마만큼 효과적으로 대처할 수 있는지 알아보았다. 위와 같은 실험에서 지능캐릭터의 에너지가 부족할 경우 에너지를 회복하여 대처하는지를 관찰한 결과 우리의 예측대로 행동하는 것을 볼 수 있었다. 이런 결과를 볼 때 본 논문에서 제안한 에너지 개념을 도입한 지능 캐릭터가 제대로 학습하고 에너지양에 따라서 적절한 동작을 할 수 있음을 알 수 있다.

본 논문의 구성은 다음과 같다. 2절에서는 게임 규칙에 대하여 알아보고, 3절에서는 유전자 알고리즘과 염색체 구조에 대하여 설명한다. 4절에서는 실험 결과를 검토 및 분석하고 5절에서 결론을 맺는다.

## 2. 게임 규칙

<표 1>은 본 논문에서 적용한 대전형 액션 게임의 캐릭터

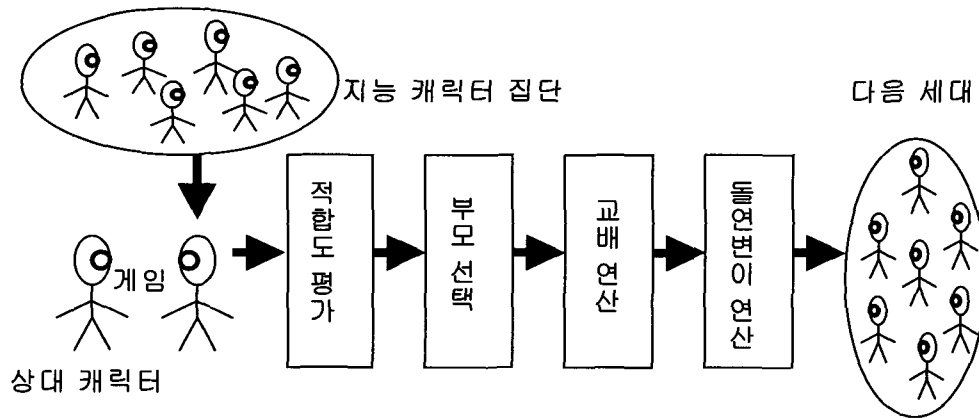
<표 1> 행동 종류, 유효거리, 단계 및 점수

	행동 종류	유효거리	단계	점수
0	정지(ID)	-	1	0
1	진진(GO)	-	1	0
2	후진(BK)	-	1	0
3	점프(JP)	-	1	0
4	앉기(DN)	-	1	0
5	막기(GD)	-	1	0
6	아래 주먹(DP)	0 - 4	2	1
7	위 주먹(UP)	0 - 4	3	2
8	아래 발 공격(DA)	3 - 7	4	3
9	위 발 공격(UA)	3 - 7	5	4
10	옆 차기(SK)	6 - 10	6	5
11	돌려 차기(RK)	6 - 10	7	6
12	들어 찌기(LC)	9 - 13	8	7
13	돌려 찌기(RC)	9 - 13	9	8
14	특수 공격1(SP1)	0 - 4	3	3
15	특수 공격2(SP2)	3 - 7	5	5
16	특수 공격3(SP3)	6 - 10	7	7
17	특수 공격4(SP4)	9 - 13	9	9

행동 종류별 유효거리 및 행동을 종료하는데 소요되는 단계와 점수를 보여주고 있다.

<표 1>에서 정지에서 막기까지는 공격이 아닌 이동과 방어행동으로 유효거리가 필요 없으며 수행하는데 필요한 단계도 가장 작은 값으로 1을 주었다. 이러한 행동은 공격이 아니므로 점수를 획득하지 못한다. 다만 공격을 할 수 없는 상황에서 이동 행동으로 유효거리를 벗어나거나 막기를 통하여 피해를 최소화 하는데 사용된다. 아래 주먹 공격부터 돌려 찌기까지는 일반기술이고 SP1에서 SP4까지가 특수기술이다. 일반 기술 공격은 동시에 성공할 경우 모두 점수를 인정하였으며 방어 행동을 하면 점수를 인정하지 않았다.

에너지는 온라인 게임을 포함하여 최근의 컴퓨터 게임에 주로 사용되는 두 종류의 에너지를 도입하였다. 하나는 Health Point(HP)로 생명력을 의미하며 이 에너지 값은 공격을 하는 캐릭터는 변화가 없는 반면 공격을 당하는 캐릭터에서 주어진 양만큼 감소한다. 다른 하나는 Mana Point(MP)로서 특수기술을 사용하는 캐릭터에서 해당 양만큼 감소하며 공격을 당하는 캐릭터의 HP 또한 해당 양만큼 감소한다. 일반 공격은 상대가 방어 행동을 하면 획득하는 점수는 없으나 특수기술(SP1~SP4)은 상대가 방어 행동을 하더라도 점수를 획득하는 것으로 하였다. <표 1>에서 점수는 공격을 받은 경우 에너지가 감소(HP)되는 양을 나타내며 특수 공격(SP1~SP4)은 공격을 받는 쪽의 에너지도 감소하지만 공격을



(그림 1) 유전자 알고리즘의 개요

하는 쪽의 에너지(MP)도 소모하게 된다. 특수기술은 주어진 거리에서 일반기술보다 더 많은 타격을 상대 캐릭터에게 줄 수 있으나 공격을 한 캐릭터의 MP도 감소하게 됨으로 남아 있는 MP를 감안하여 사용해야 한다. 게임의 승패는 일정시간 대결 후 남은 HP양으로만 결정하기 때문에 MP의 경우 많이 남기는 것 보다는 특수기술을 적절히 배합하여 상대의 HP를 많이 감소하게 하는 것이 좋다. 일정 시간 대결 후 지능 캐릭터의 HP가 0이 아닌데 상대 캐릭터의 남은 HP가 0이면 지능 캐릭터의 KO승으로 결정한다. 두 캐릭터의 HP가 모두 0이 아닌 경우 남은 HP로 승패를 결정하며 (판정승) HP가 0이던 아니던 같은 경우 비긴 것으로 한다<sup>1)</sup>. 보다 게임 규칙을 일반화하기 위하여 현재 온라인 게임에서 많이 사용되는 에너지 회복기능도 적용하였다. 이 기능은 일정기간동안 상대 캐릭터의 공격으로부터 점수를 잃지 않을 경우 에너지를 회복하는 기능으로 본 논문에서는 일정시간 동안 점수를 잃지 않을 경우에 HP와 MP를 초기 점수(100점)의 3%(3점) 충전되도록 하였다. 여기서 점수를 잃지 않는다는 것은 상대방의 공격을 막거나 아니면 이동 행동으로 유효거리 밖으로 이동하여 점수를 잃지 않는 방법이다. 이와 같이 에너지 회복기능을 적용함으로써 지능 캐릭터가 자신의 에너지가 적을 경우 무조건 공격 행동만 하는 것이 아니라 이동이나 방어 행동을 적절히 하여 자신의 HP와 MP 에너지를 충전하는 보다 전략적인 행동 양식을 선택하게 된다.

### 3. 유전자 알고리즘과 염색체 구조

본 논문에서 에너지 개념을 도입한 대전형 액션 게임의 지능 캐릭터 구현에 적용한 유전자 알고리즘[7, 8]은 (그림 1)과 같다. (그림 1)에서 보듯이 지능 캐릭터의 적합도를 평가하기 위하여 무작위로 생성된 상대 캐릭터 100개와 지능 캐릭터가 대결을 하게하고 그 승률을 적합도로 사용한다. 적합도에 따라 다음 세대를 생성할 부모 염색체를 선택하며 선택 방법으로는 룰렛 휠 선택(Roulette Wheel Selection)

방법[9]을 적용하였다. 선택된 부모 염색체 2개에 대해서 블록 교배(Block Crossover)[10]를 사용하여 다음 세대를 생성한 후 돌연변이[11] 연산을 거치게 된다. 이와 같은 과정을 종료조건이 만족할 때까지 반복하게 하였다. 염색체는 캐릭터간의 거리와 상대 캐릭터의 행동에 따라 지능 캐릭터가 어떤 행동을 할 것인지를 규정한다. 지능캐릭터에 HP와 MP 에너지를 적용하기 위해서는 현재의 행동과 단계, 거리, 과거행동을 나타내는 구조에 에너지 부분을 추가하여야 한다. 그러므로 추가되는 HP와 MP 에너지는 새로운 차원의 파라미터이기 때문에 그 개수가 커지면 염색체의 크기가 매우 커지게 된다. 염색체가 커지면 현재 상황에 대한 보다 세부적인 결정을 내릴 수 있는 장점이 있으나 학습시간을 길게 하고 상황에 따라서 너무 상황별로 민감한 결정을 하게 되어 오히려 단점으로 작용할 수도 있다.

본 논문에서는 염색체 크기에 영향을 미치는 에너지의 양의 크기는 에너지 개념을 도입하는 의미가 있게끔 (그림 2)와 같이 HP와 MP를 5개의 영역(0~20, 21~40, 41~60, 61~80, 81~100)으로 구분하여 표현하였다. 결국 HP와 MP로 표현되는 에너지의 가짓수는 AA부터 EE까지 모두 25가지이다.

(그림 3)은 에너지 HP와 MP를 포함하는 염색체 구조를 보여주고 있다. (그림 3)에서 맨 위의 표는 거리와 행동의 단계에 따른 염색체의 표현이며 중간에 있는 표는 과거 행동을 추가한 염색체의 표현이고 맨 아래 있는 표는 에너지 개념을 도입한 염색체의 표현이다. 중간 표나 맨 아래 표는 바로 위에 있는 표의 모든 항에 들어가는 것을 대표해 그린 그림으로서 중간 표의 지능 캐릭터의 행동은 에너지 조합수(25)로 맨 아래와 같이 다시 나누어진다. 그러므로 염색체의

H P	A	B	C	D	E
	0-20	21-40	41-60	61-80	81-100
M P	A	B	C	D	E
	0-20	21-40	41-60	61-80	81-100

(그림 2) HP와 MP의 구조

1) 이와 같은 게임 규칙은 기존의 온라인 게임이나 컴퓨터 게임을 고려하여 결정된 것으로서 어느 정도 일반성이 있다.

현재행동	I D	G O	B K	J P	D N	G D	주먹 공격						...	SP1	SP2	SP3	SP4
							DP		UP								
거리	소요단계	1t	1t	1t	1t	1t	1t	1t	2t	1t	2t	...	3t	5t	7t	9t	
0		8	6	2	6	3	6	11	0	7	4	2	...	2	7	9	3
1		7	3	5	9	10	9	2	4	9	2	1	...	8	1	0	12
2		6	9	2	1	10	1	8	2	6	9	3	...	5	9	13	9
⋮		⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮
12		3	13	7	4	9	5	11	2	0	10	4	...	6	1	15	3
13		7	4	9	3	2	4	9	12	0	4	9	...	16	8	4	8

과거행동	I D	G O	B K	J P	D N	G D	D P	U P	D A	U A	S K	R K	L C	R C	SP1	SP2	SP3	SP4
지능 캐릭터행동	9	6	12	6	3	0	7	11	4	17	13	4	8	2	16	3	1	10

조 합	AA (1)	AB (2)	AC (3)	AD (4)	AE (5)	BA (6)	.....	EA (21)	EB (22)	EC (23)	ED (24)	EE (25)
지능 캐릭터행동	9	6	12	6	3	0	.....	4	8	2	16	4

(그림 3) 염색체 구조의 일부

총 항은 상대 캐릭터와의 (거리)\*(상대캐릭터의 현재 행동의 단계의 합)\*(상대 캐릭터의 과거의 행동의 종류)\*(에너지조합)으로 총 466,200(14\*74\* 18\*25)개 이다.

각 항별로 지능 캐릭터가 수행할 행동18개중 하나가 저장된다. 그러므로 총 탐색공간은 18의 466,200승이 되어 거의 무한대에 가깝게 되나 실제로 게임 규칙에 따른 총 탐색공간은 이보다 훨씬 작다. 작은 이유로는 각각의 거리에서 유효공격의 수가 제한되어 있기 때문이다. 유효 탐색 공간이 비록 총 탐색 공간보다 작지만 매우 크기 때문에 전역 최적해를 찾는 것은 매우 어려우며 매우 많은 시간이 필요하다.

#### 4. 실험 결과 및 검토

에너지 개념을 도입한 지능 캐릭터의 성능을 평가하기 위하여 세 가지로 구분하여 지능 캐릭터의 성능을 평가하였다. 첫 번째 실험에서는 개체가 세대가 진행됨에 따라서 진화하는 지를 알아보기 위하여 1세대의 게임 후에 가장 좋은 개체의 지능 캐릭터와 상대 캐릭터<sup>2)</sup>의 HP와 MP양 과 마지막 세대(100% 승리하는 개체)에서 지능 캐릭터와 상대 캐릭터의 HP와 MP양을 비교 하였다. 두 번째 실험에서는 지능

캐릭터의 학습이 일반성을 갖는지를 알아보기 위한 실험으로 학습에 이용된 상대 개체가 아닌 새로운 상대 개체와의 대결을 수행해 본다. 즉 1세대에서 가장 좋은 승률을 보인 염색체와 새로 생성된 상대 캐릭터와 게임을 하여 승률을 알아보았으며 100% 승률을 보인 마지막 세대에서 가장 좋은 개체와 새로 생성된 상대 캐릭터와 게임을 하여 승률을 비교하였다. 세 번째 실험에서는 승률이 높은 개체가 낮은 개체보다 실제로 더 좋은지 알아보기 위해서 59%, 71%, 80%, 90%, 100%인 개체를 선택하여 각각 서로 대결을 해 보았다.

##### 4.1 실험 결과

유전자 알고리즘에서 다음 세대 선택을 위한 적합도로는 지능 캐릭터가 무작위로 생성된 상대 캐릭터 100개와 Max\_Tick 동안 게임을 한 결과의 승률을 적합도로 적용하였다. 실험에 사용된 파라미터 값은 교배확률=1, 돌연변이 확률=0.0002, 개체크기=50, HP와 MP의 초기 에너지는 100, 한 번의 게임을 하는데 소요되는 시간(Max\_Tick)은 1500, 일정시간 상대로 부터 공격을 받아 점수를 잃지 않는 기간(No\_Damage\_Tick) 등이다. 여기서 Max\_Tick의 의미는 두 캐릭터가 한 번의 게임을 하는데 1500틱이 소요됨을 의미하며 No\_Damage\_Tick 은 캐릭터가 상대방의 공격을 받더라도 이동이나 막기 행동을

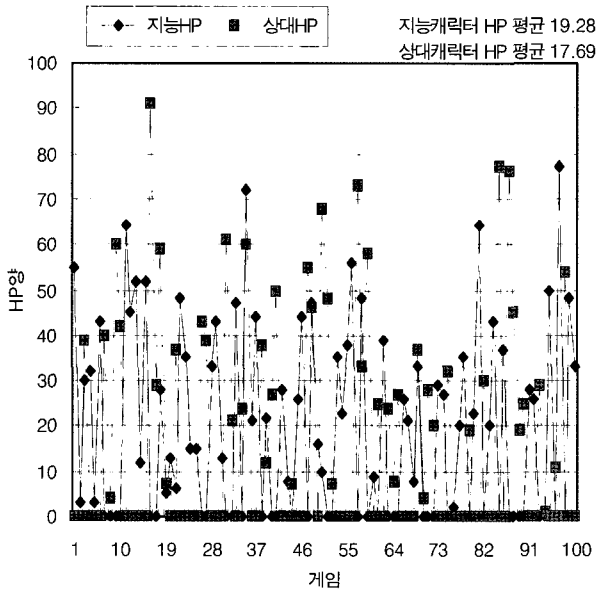
2) 상대 캐릭터는 특정 개수만큼 무작위 적으로 생성된다.

<표 2> 실험에 사용한 파라미터 값

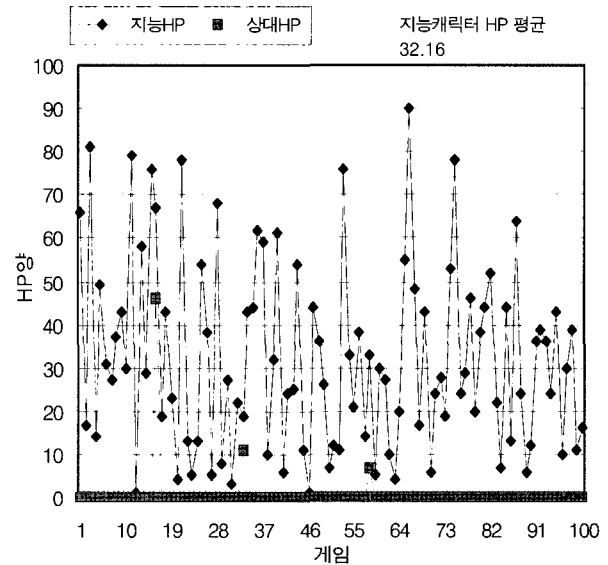
No.	SEED	No_Damage_Tick	HP,MP 증가	세대수
1	99	20	3	611
2	99	30	3	2816
3	199	40	5	141

<표 3> 1세대와 마지막세대의 가장 좋은 개체가 새로운 캐릭터와의 게임 결과

구분	세 대		
	A(1세대)	B(611세대)	
HP의 양	지능 캐릭터	18.82	25.76
	상대 캐릭터	16.51	14.18
지능 캐릭터의 승률		53%	67%



(그림 4) 1세대 후의 HP 비교



(그림 5) 마지막 세대(611)에서의 HP 비교

하여 20틱 동안 점수를 잃지 않을 경우에 HP와 MP의 에너지를 3점이 회복되는 기간을 의미한다. 이와 같은 파라미터 조합으로 실험한 결과 <표 2>와 같이 모두 3번의 실험에서 각각 세대수가 611, 2816, 141에서 100%의 승률을 보였다. 이 후 실험결과 및 검토에서는 1번 실험 결과만으로 설명하기로 한다.

(그림 4)는 첫 번째 실험 결과로서 1세대와 마지막 세대의 지능 캐릭터와 상대 캐릭터의 HP양을 비교한 그래프이다. 1세대에서 가장 좋은 결과를 보인 47번 개체의 지능캐릭터와 상대 캐릭터의 HP양을 보여주고 있다. 지능 캐릭터의 HP 평균은 19.28이고 상대 캐릭터의 평균은 17.69로 지능 캐릭터가 약간 더 높게 나타났다. 1세대는 진화가 이루어지기 전의 결과로서 비교에는 의미가 없으나 마지막 세대와의 비교를 위하여 표시하였다. 1세대 중에서 가장 좋은 개체(47번)의 승률은 53%였으며, 이때 지능 캐릭터의 KO율은 46%를 보였다. (그림 4)에서 보듯이 1세대에서는 지능캐릭터가 학습을 하지 않은 관계로 게임에 따라서 지능캐릭터의 HP 잔량이 많은 경우도 있고 상대캐릭터의 HP 잔량이 많은 경우도 있다. 평균적으로 지능 캐릭터의 HP 잔량이 조금 많으나 이것은 무작위 값에 따라서 반대가 될 수 있는 것으로 큰 의미는 없다.

(그림 5)는 100%의 승률을 보인 611세대의 4번 개체의 지능 캐릭터와 상대 캐릭터의 HP 양을 보여주고 있다. 그래프

에서 보는 바와 같이 지능 캐릭터의 HP양이 상대 캐릭터보다 월등히 많음을 알 수 있으며 이때 지능 캐릭터의 HP 평균은 32.16이고 상대 캐릭터의 HP 평균은 0.64로 지능 캐릭터가 더 높게 나타났다. (그림 4)와 (그림 5)를 비교해 보면 지능 캐릭터의 HP양은 더 증가했고 상대 캐릭터의 HP양은 감소하였다. 이는 지능 캐릭터가 주어진 환경에 적응하면서 게임 규칙을 잘 학습했음을 보여준다. 마지막 세대에서 가장 좋은 개체(4번)의 승률은 100%였으며, 이때 지능 캐릭터의 KO율은 97%를 보였다. (그림 5)에서 상대 캐릭터의 HP가 0인 경우 지능 캐릭터의 KO승을 의미한다.

<표 3>은 두 번째 실험 결과로서 100%의 승률을 보인 개체가 새롭게 무작위로 선택된 다른 개체 다시 말해서 학습하지 않은 개체에 대해서도 효과가 있는지 알아보았다. <표 3>의 A는 1세대에서 가장 좋은 성능(53%승)을 보인 47번 개체가 새로 생성된 상대 캐릭터와 게임을 한 결과이고 <표 3>의 B는 마지막 세대(611)에서 100%승률을 보인 4번 개체와 새로 생성된 상대 캐릭터와 게임을 한 결과이다. B의 경우에 좀더 나은 승률(67%)을 보여주고 있으나 그 차이(14%)는 그리 크지 않다. 이와 같이 승률이 현저하게 차이가 나지 않는 이유는 지능 캐릭터의 학습 대상인 무작위로 행동하는 캐릭터 100개의 행동 패턴이 게임에서 발생할 수 있는 게임 상황에 비해 매우 작기 때문이다. 즉 비록 지능 캐릭터가 특정한 100개의 무작위 행동을 하는 캐릭터의 행동

〈표 4〉 각 승률별 개체를 각각 대결한 결과

A	B	승	무승부	패
100%	59%	32	0	18
90%	59%	28	2	20
80%	59%	26	1	23
80%	71%	27	2	21

패턴에 적절히 대응하는 방법을 학습했다 하더라도 또 다른 100개의 무작위 행동을 하는 캐릭터의 행동 패턴이 학습한 행동패턴과 많이 겹치지 않는 한 지능 캐릭터는 새로운 행동 패턴에 적절한 대응을 할 수 없다는 것이다. 이는 게임에서의 행동패턴 탐색 공간이 매우 크기 때문에 발생하는 것으로서 무작위로 행동하는 상대캐릭터의 개수를 크게 하면 완화될 수 있으나 학습시간이 매우 커지는 문제점이 있다. 실제 게임에 응용할 경우에는 학습시간과 학습범위를 적절히 고려하여 적당한 개수의 상대캐릭터를 선택해야 한다.

세 번째 실험에서는 승률이 59%, 71%, 80%, 90%, 100%인 개체를 선택하여 각각 대결 한 결과를 <표 4>에 표시하였다. 여기서 59%, 71%의 승률을 선택한 이유는 유전자 알고리즘의 실행단계에서 60%와 70%의 승률을 건너뛰었기 때문이다. 즉 59%의 승률 다음에 63%의 승률을 보였기 때문에 60%에 근접한 59%의 승률을 선택하여 적용하였다. 표에서 승과 패는 A와 B의 대결에서 A의 결과이다. 본 실험에서 개체 크기는 50으로 하였으므로 모두 50번의 대결이 이루어졌으며 승률이 높게 학습된 개체가 다른 개체에 대해서도 효과가 있는지 알아보았다. 실험 결과 <표 4>와 같이 승률이 높은 개체가 낮은 개체보다 승률이 높음을 알 수가 있었다. 이런 결과로 볼 때 비록 무작위로 행동하는 100개의 캐릭터의 행동 패턴이 전체 행동 패턴에 비해 적은 영역을 차지하더라도 해당 행동 패턴에 대해 많이 학습하는 것이 일반적으로 좋을 수 있다.

4.2 실험 분석 및 검토

(그림 6)은 지능 캐릭터와 상대 캐릭터의 행동비교로서 에너지양이 적을 경우 두 캐릭터의 행동을 분석하기 위한 그림이다. 상대 캐릭터가 415틱에서 KO패를 한 게임에서 마지막 부분의 일부만을 보여주고 있다. 여기서 player가 상대 캐릭터이며, com이 지능 캐릭터의 행동이다. 상대 캐릭터의 행동을 보면 MP 양이 적어 특수공격(SP1-SP4) 하더라도 점수로 인정되지 않는데도 특수공격을 여러 번(3번) 하는 것을 볼 수가 있다. 그러나 지능 캐릭터는 MP양이 적을 경우에 이동이나 방어 행동 혹은 적절한 행동을 하여 389틱에서 에너지를 충전하는 것을 볼 수가 있다. 이와 같이 HP나 MP 에너지가 적을 경우에는 무조건 공격만을 하지 않고 일정시간 동안 방어나 이동 행동을 하여 에너지를 충전한 후에 다시 게임에 임하는 것을 확인하였다.

에너지 회복기능과 관련하여 지능캐릭터의 행동방식을

```

tick0372 : com action 0, hp=8, mp=2, dist=9
tick0373 : com action 9, hp=8, mp=2, dist=9
tick0375 : player action 17, hp=13, mp=3
tick0378 : com action 6, hp=8, mp=2, dist=9
tick0379 : player action 0, hp=13, mp=3
tick0380 : player action 9, hp=13, mp=3
tick0380 : com action 8, hp=8, mp=2, dist=9
tick0384 : com action 2, hp=8, mp=2, dist=9
tick0385 : player action 6, hp=13, mp=3
tick0385 : com action 8, hp=8, mp=2, dist=10
tick0387 : player action 16, hp=13, mp=3
tick0389 : com action 10, hp=11, mp=5, dist=10
tick0390 : player action 13, hp=13, mp=3
tick0395 : player action 4, hp=8, mp=3
tick0395 : com action 1, hp=11, mp=5, dist=10
tick0396 : player action 17, hp=8, mp=3
tick0396 : com action 2, hp=11, mp=5, dist=9
tick0397 : com action 7, hp=11, mp=5, dist=10
tick0400 : com action 0, hp=11, mp=5, dist=10
tick0400 : player action 9, hp=8, mp=3
tick0401 : com action 12, hp=11, mp=5, dist=10
tick0405 : player action 7, hp=8, mp=3
tick0408 : player action 6, hp=8, mp=3
tick0409 : player action 5, hp=1, mp=3
tick0409 : com action 11, hp=11, mp=5, dist=10
tick0410 : player action 9, hp=1, mp=3
tick0415 : player action 9, hp=1, mp=3
    
```

(그림 6) 상대 캐릭터와 지능 캐릭터의 행동 비교

살펴보기 위하여 여러 번의 실험을 하고 관찰하였다. 관찰결과 일정기간 동안 점수를 잃지 않았을 때 에너지가 회복되는 시간인 No\_Damage\_Tick을 작게 하면 상대 캐릭터나 지능 캐릭터의 에너지 회복이 빈번히 발생하여 KO율이 적었으며 주어진 시간 동안 게임을 한 후 판정승으로 승패가 결정되는 경우가 많았다. 반면에, No\_Damage\_Tick을 크게(60 이상) 하면 에너지의 회복이 전혀 없어 No\_Damage\_Tick을 적용하는 효과를 보지 못하였다. 일정 시간 동안 게임이 이루어지는 Max\_Tick 값이 크면 클수록 KO율이 증가하였으며 일정기간 동안 상대 공격로부터 점수를 잃지 않으면 에너지가 회복되는데 이때 충전되는 에너지의 양이 적으면 KO율이 증가하였다. 이러한 결과는 우리의 직관과 일치하는 결과이다.

본 실험을 통하여 에너지 개념을 도입하는 지능 캐릭터는 환경이 바뀌더라도 잘 적응하는 것을 확인하였으며 승률이 높은 개체가 일반적인 환경에서도 높은 승률을 보였다. 또 HP나 MP 에너지가 적을 경우에는 무조건 공격만을 하지 않고 일정시간 동안 방어나 이동 행동을 하여 에너지를 충

전한 후에 다시 게임에 임하는 것을 확인하였다.

### 5. 결 론

본 논문에서는 대전형 액션 게임의 지능 캐릭터를 구현하는데 있어서 보다 실제적인 구현을 위하여 에너지 개념을 도입하였다. 에너지는 현재 온라인 게임에서 많이 사용하고 있는 HP 와 MP를 사용하였으며 일정시간 동안 상대 공격을 방어하거나 이동하여 회피한 경우에는 에너지가 회복하는 기능도 구현하였다.

에너지 개념을 적용한 지능 캐릭터가 제대로 학습하는지 학습의 효과가 있는지 얼마나 많은 정보를 학습하는지를 알아보기 위하여 여러 종류의 실험을 하였다. 실험 결과 대부분의 실험에서 무작위로 행동하는 상대 캐릭터에 대하여 100% 승률을 보이는 개체로 학습함을 보였고 또한 지능 캐릭터를 상대 캐릭터로 하여 학습한 경우 무작위 상대캐릭터의 경우보다 훨씬 빠르게 학습하는 것을 볼 수 있었다. 또한 에너지가 적은 경우 방어와 이동을 통하여 상대의 공격을 회피하며 에너지를 회복하여 대응하는 것을 볼 수 있었다. 이런 결과는 본 논문에서 제안한 에너지를 적용한 지능 캐릭터가 효과적으로 상대방에 대응할 수 있다는 것을 보여준다. 앞으로 본 논문에서 제안한 방법을 실제 게임에 적용해 보는 일이 필요하다 하겠다.

### 참 고 문 헌

[1] D.C Pottinger and J.E.Laird, "Game AI: The State of the Industry Part 2," Gameasutra Magazine, Vol.8, Nov., 2000.  
 [2] S. Woodcock, "Game AI: The State of the Industry," Gameasutra Magazine, Vol.1, 2000.  
 [3] 이만재, "게임에서의 인공지능 기술," 한국정보처리학회지, Vol.9, No.3, pp.69-76, May, 2002.  
 [4] 이면섭, 조병현, 성영락, 정성훈, 오하령, "유전자 알고리즘을 이용한 대전형 액션 게임의 지능 캐릭터", 한국 정보처리학회 논문지, Vol.12-B, No.3, pp.329-336, June, 2005.  
 [5] 이면섭, 조병현, 성영락, 정성훈, 오하령, "대전 액션 게임에서의 유전자 알고리즘을 이용한 지능 캐릭터 구현", 대한 전자공학회 논문지, Vol.41-TE-4, pp.119-128, Dec., 2004.  
 [6] 이면섭, 조병현, 성영락, 정성훈, 오하령, "상대방의 행동 패턴에 적용하는 지능 캐릭터의 구현", 대한 전자공학회논문지, Accepted.  
 [7] I.Faybish, "Applying the Genetic Algorithm to the Game of Othello," VRIJE University of BRUSSEL, 1999.  
 [8] D. Whitley, R. Beveridge, C. Guerra and C. Graves. "Messy Genetic Algorithms for Subset Feature Selection", International Conference on Genetic Algorithms 1997.  
 [9] Darrell Whitley, "A Genetic Algorithm Tutorial", Statistics

and Computing Vol.4, pp.65-85 1994.

[10] C.Anderson, K.Johns and J.Ryan, "A two-dimensional genetic algorithm for the Ising problem," Complex Systems, Vol.5, pp.327-333, 1991.  
 [11] Darrell Whitley "An Overview of Evolutionary Algorithms" Journal of Information and Software Technology. Vol.43, pp. 817-831, 2001.



#### 이 면 섭

e-mail : leems@icc.ac.kr  
 1985년 국민대학교 전자공학과(공학사)  
 1987년 인하대학교 전자공학과(공학석사)  
 2005년 국민대학교 전자공학과(공학박사)  
 1990년~현재 인천전문대학 컴퓨터  
 정보과 전임강사, 조교수, 부교수

관심분야: 유전자 알고리즘, 게임, 컴퓨터 그래픽



#### 조 병 현

e-mail : bhcho@etri.re.kr  
 1997년 국민대학교 전자공학과(공학사)  
 1999년 국민대학교 전자공학과  
 (공학석사)  
 2005년 국민대학교 전자공학과(공학박사)  
 2005년~현재 전자통신연구원 디지털  
 콘텐츠연구단 게임기술개발센터  
 네트워크 가상환경연구팀 연구원

관심분야: 유전자 알고리즘, 시뮬레이션, 신경망



#### 정 성 훈

e-mail : hjung@hansung.ac.kr  
 1988년 한양대학교 전자공학과(공학사)  
 1991년 한국과학기술원 전기 및  
 전자공학과(공학석사)  
 1995년 한국과학기술원 전기 및  
 전자공학과(공학박사)

1995년~1996년 한국과학기술원 전기및전자공학과(위촉연구원)

1996년~1998년 한성대학교 정보전산학부 정보통신공학전공  
 전임강사

1998년~2002년 한성대학교 정보전산학부 정보통신공학전공  
 조교수

2002년~현재 한성대학교 정보통신공학과 부교수

관심분야: 신경망, 뉴로퍼지, 유전자알고리즘, 시스템 생물학



### 성 영 락

e-mail : yeong@kookmin.ac.kr  
1989년 한양대학교 전자공학과(공학사)  
1991년 한국과학기술원 전기 및  
전자공학과(공학석사)  
1995년 한국과학기술원 전기 및  
전자공학과(공학박사)

1995년~1996년 한국과학기술원 위촉연구원  
1996년~1998년 국민대학교 전자공학부 전임강사  
1998년~2002년 국민대학교 전자공학부 조교수  
2002년~현재 국민대학교 전자정보통신공학부 부교수  
관심분야: 시뮬레이션, 고장감내, 내장형 시스템



### 오 하 령

E-mail:hroh@kookmin.ac.kr  
1983년 서울대학교 전기공학과(공학사)  
1983년~1986년 삼성전자 종합연구소  
1988년 한국과학기술원 전기전자과  
컴퓨터공학전공(공학석사)  
1992년 한국과학기술원 전기전자과  
컴퓨터공학전공(공학박사)

1992년~1996년 국민대학교 전자공학부 조교수  
1996년~2001 국민대학교 전자공학부 부교수  
2001년~현재 국민대학교 전자정보통신공학부 교수  
관심분야: 병렬처리, 내장형 시스템, 고장감내