



An Implementation of Shooting Games using Genetic Algorithms

Myun-Sub Lee*

Department of Computer Science and Engineering, Incheon University

ABSTRACT

This research is an implementation of a shooting game that shooting down the opposite term's fighter plane by firing bullets while moving our term's fighter plane. The opposite team's fighter plane is created randomly and perish when it gets out of the realm. There are two offense strategies . The first one is attacking while moving oneself's plane. The other strategy is that the player transfer his plane but offense is conducted by genetic algorithms. The player can move their plane in 3 ways: left, right, and stop. 100 opposite fighter planes are created randomly. The planes gotten out of the realm is excluded. The number of shot down plane in the realm and accuracy rate was evaluated. As a result, satisfied result was shown in second experiment-applying genetic algorithms.

© 2014 KKITS All rights reserved

KEYWORDS: Genetic Algorithms, Artificial Intelligence, Game Characters, Shooting Games, Chromosomes

ARTICLE INFO: Received 17 March 2014, Accepted 11 April 2014.

1. 서론

*Corresponding author is with the Department of Computer Science & Engineering, Incheon University, (Songdo-dong)119 Academy-ro, Yeonsu-gu, Incheon, Korea

E-mail address: nantian@incheon.ac.kr

슈팅게임은 게이머가 다양한 종류의 무기를 발사하여 적을 공격하거나 적의 공격을 피해 가면서 게임을 진행해 나가는 방식이다. 자신을 향해 쏟아져 오는 탄환을 보고 손을 움직여 키보드 등을 조작하여 빨리 피한 후에 상대방을 공격하는 게임 방식이므로 높은 순발력을 요구한다[1-2]. 그러므로 슈팅게임은 액션게임보다도 더욱 원초적인 재미를 추구하기 때문에 스토리 같은 요소는 배제되고 킬

링 타임용으로 플레이하게 되는 것이 대부분이다. 이는 게임의 초창기부터 개발된 것으로 1962년 스페이스 워가 발표된 이후 지금까지 여러 가지 형태로 개발되어 발표되고 있다. 그러나 많은 경우 슈팅게임의 난이도와 흥미는 캐릭터 행동방식과 개발 알고리즘에 따라서 크게 좌우된다. 즉, 캐릭터의 행동 제어 알고리즘이나 캐릭터의 지능레벨에 따라 게임의 난이도와 흥미가 크게 의존된다. 그리고 슈팅게임의 경우 여러 개의 상대 캐릭터가 화면 내에 존재하게 되고 어떤 캐릭터는 상대방에 의해 파괴되더라도 새로운 캐릭터가 화면에 나타나는 것이 일반적이다[3-4]. 그러나 게임의 흥미를 유발하기 위해서는 난이도의 향상에 신경을 써야 한다. 실제 슈팅 게임에서 난이도는 플레이어의 조작 기술에 따라 커다란 차이를 보이고 있다[5]. 그러므로 플레이어가 직접 조작하지 않고 각 캐릭터가 인공지능으로 동작하도록 하여 게임의 난이와 흥미를 유발할 필요가 있다.

본 연구에서는 게임의 흥미를 유발하고 제안한 방법의 성능 평가를 위해 두 가지로 실험하였다. 첫 번째는 자신의 객체 이동과 공격을 플레이어에 의해 조작되도록 하였으며 두 번째는 아군의 이동은 플레이어가 조작하고 공격은 유전자 알고리즘을 적용하여 유리한 위치일 때 공격 하도록 하였다. 이것은 자신의 캐릭터는 존재할 확률이 가장 높은 위치에서 상대 캐릭터에 대해 공격무기를 발사하기 위해서이다.

본 논문의 구성은 다음과 같다. 2장에서는 유전자 알고리즘에 대해 알아보며 3장에서는 슈팅게임의 규칙과 방법에 대하여 설명하며, 4장에서는 제안한 방법의 평가 및 검토이며 5장에서는 결론을 맺는다.

2. 유전자 알고리즘

유전자 알고리즘(Genetic Algorithm)이란 자연계 생물의 유전과 진화의 메커니즘을 공학적으로 모델화하는 것에 의해 생물의 환경 적응 능력과 자연도태 원리를 기반으로 만들어진 계산모델이다[6-7]. 유전자 알고리즘에서 하나의 개체는 <그림1>과 같이 하나의 염색체로 표현한다. 염색체의 표현에는 대부분 2진수로 표현하지만 10진수나 문자를 사용하는 경우도 있다.

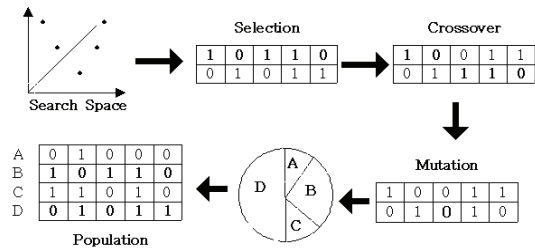


그림 1 유전자 알고리즘의 흐름
Figure 1. the flow of genetic algorithm

유전자 알고리즘의 진행 과정은 다음과 같다. 랜덤하게 N개의 초기 염색체를 생성한다. 초기 집단 염색체를 만들 때에는 난수를 이용하여 여러 개의 염색체를 생성한다. 생성된 염색체 집단중에서 각 개체의 적합도를 계산한다. 적합도가 좋은 개체 2개를 선택하여 교배시켜 좋은 염색체가 다음 세대에 전달 할 수 있도록 한다. 선택 과정은 자연도태를 모델링한 것으로 적합도에 따라서 개체를 다음 세대로 전달하기도 하고 도태시키기도 한다. 선택 방법에는 가장 일반적인 토너먼트 선택을 적용하였다.

선택 연산 방법 중에서도 가장 기본적인 방식으로 룰렛-휠 선택(Roulette-Wheel Selection)이며 <표 1>과 같이 적합도에 철저하게 비례하여 선택된다[8]. 즉 적합도가 큰 개체일수록 선택되기 쉽고, 다른 개체와 교배할 수 있는 가능성이 높아진다. 평

조작하고 공격은 유전자 알고리즘을 적용하여 유리한 위치일 때 공격 하도록 하였다. 이 아군기의 위치는 화면 아래에 있으며, 공격은 위 방향으로 한다.

또 하나의 객체는 적군기로서 화면 상단에서 좌, 우로 이동하면서 아래 방향에 있는 아군기를 향하여 무기를 랜덤하게 발사하여 공격을 가하는 전투기로 하나의 화면에 5개의 객체가 존재하도록 하였다. 본 연구에서 슈팅 게임에 적용한 규칙은 다음과 같다.

1. 게임 화면은 x, y 좌표의 최대값을 x_{max} , y_{max} 로 한 경우로 표현되는 직각좌표 영역이고 원점 (0,0)은 좌측상단으로 한다.

$$0 \leq x < x_{max},$$

$$0 \leq y < y_{max},$$

2. 게임 화면의 모든 움직임은 같은 단위시간으로 제어되고 이것을 스텝(단계)이라 한다.

3. 게임 화면에는 2개의 객체 즉 아군기(P기)와 적군기(E기)가 존재한다.

P기 : 아군기의 조작은 플레이어에 의해 이동과 공격이 이루어지는 방법과 유전자 알고리즘에 의해 이동과 공격이 이루어지도록 2가지로 적용하였다. 전투기의 이동은 방향키 한 번에 한 스텝 이동한다.

E기 : P기에 대해서 랜덤하게 공격을 하는 적기로 적기의 행동 제어 알고리즘은 1 스텝 마다 이루어진다. E기는 위치는 x_{ran} , $y_{max}-1$ 로 하였다. 여기서 x_{ran} 는 랜덤한 좌표 값이다.

4. 적기로부터의 아군기의 공격은 아군기를 향하여 공격한다. 공격탄의 발사방향은 아군기가 있는 아래 방향으로 수직낙하하며 공격탄은 이 최하단에 도달하게 되는 x좌표로 설정하였다. 공격탄은 발사로부터 y_{max} 스텝 후에 아군기가 위치해 있는 최하단에 도달하는 것으로 하였다.

5. 각 스텝마다 아군기와 적기의 공격탄이 같은

좌표가 되는 경우는 자신의 전투기(P기)는 파괴된다. 자신의 전투기는 게임 화면 밖으로 나갈 수 없으며, 화면 밖으로 나간 공격탄은 소멸되는 것으로 하였다.

현재부터 y_{max} 스텝 사이의 아군기의 이동상황을 L(좌), R(우), S(정지)로 표현하고, 이 때의 염색체는 y_{max} 로 그림 3과 같이 문자열로 표현한다. 염색체가 표시하는 아군기의 이동상황과 화면상에 존재하는 공격탄(기존 적기에서 발사된 공격탄)의 정보로부터 그 염색체가 가지는 이동상황의 안전도를 다음과 같이 계산한다.

L	R	L	S	R	L	S	S	L	L	L	S	S	L
---	---	---	---	---	---	---	---	---	---	---	---	---	---

그림 3. 염색체의 구조

Figure 3. chromosome configuration

(1) 매 스텝마다 적기에서 발사된 탄환과 최하단에 있는 아군기 사이의 거리를 구한다. 이 때 최하단에 적기에서 발사된 탄환이 여러개 존재하는 경우에는 아군기와 가까운 쪽의 탄환의 거리를 구한다.

(2) 현재부터 y_{max} 스텝 사이에서 각 스텝마다 적기에서 발사된 탄환과 아군기와의 거리를 구하여 이것을 이 염색체의 안전도로 한다.

(3) 아군기나 적기에서 발사된 탄환이 상대방과 접촉하면 격추된 것으로 하고 그 시간까지의 스텝 수(거리)를 염색체의 안전도로 한다. 이 계산에서 구해진 안전도의 값이 크면 클수록 염색체의 적합도가 큰 것이다.

유전자 알고리즘을 적용한 방법은 다음과 같다. 컴퓨터는 랜덤하게 이동과 공격을 하는 적군기의 이동 방향 이외의 모든 정보를 파악 가능한 것으로 한다. 현재 적군기의 위치로부터 y_{max} 스텝 후의

위치를 앞에서 설명한 적합도를 바탕으로 하여 그 방향으로 향해 탄환을 발사한다. 유전자 알고리즘을 이용하여 안전도가 높은 염색체를 선택하여 그 염색체가 나타내는 정보대로 이동하여 y_{max} 스텝 후에 적군기가 존재하는 위치로 향해 유전자 알고리즘을 적용한 아군기는 탄환을 발사한다.

일반적으로 유전자 알고리즘에서는 주어진 문제에 대해서 근사해를 구하는 것으로 종료되지만 본 연구에서는 아군기가 탄환을 발사하는 것으로 알고리즘이 종료되는 것이 아니라 다음 탄환의 발사 방향을 다시 계산할 필요가 있다. 다음에 발사될 탄환의 방향을 계산할 때 초기 개체 집단을 모두 랜덤하게 생성하는 것은 아니고 바로 전에 발사 방향을 결정 할 때 남아있던 개체 집단으로부터 다음과 같이 하여 초기 개체 집단을 생성하였다.

- (1) 이전 단계에서 선택한 방향과 같은 방향을 선택한 염색체는 남기고, 그 이외의 염색체는 제거하여 랜덤하게 다시 만든다.
- (2) (1)단계에서 제거된 염색체 자리에 다른 유전자를 하나씩 이동시켜 채워 넣는다.
- (3) 염색체의 움직임이 발생된 빈자리에 L, R, S 중에서 랜덤하게 선택하여 채워 넣는다.
- (4) 매 단계마다 적합도가 가장 좋은 염색체는 다음 세대에 반드시 전달되도록 엘리트 유전자 보존 법칙을 적용하였다. 이는 대부분의 슈팅게임에서는 리얼타임이 요구되므로 빠른 계산을 요구하기 때문이다.

4. 평가 및 검토

유전자 알고리즘의 실험 조건에서 염색체 개체 수는 50, 세대수는 연속적인 게임을 위해 무한대로 하였으며 게임 종료는 별도로 지정하였다. 염색체 길이는 14로 하였으며 염색체의 선택은 룰렛 선택

이고, 교배 방법은 블록 균인 교배 방법을 적용하였고, 교배 확률은 0.3이고, 돌연변이 확률은 0.02로 하였다. 적기 공격탄 발사 확률은 0.9에서 1 사이의 랜덤 값으로 하였다.

<그림4>는 실제 게임화면 구성이며 여기서 $\frac{P}{\langle PPP \rangle}$ 는 아군기, eYe는 적군기이고, ↑는 아군기에서 발사된 탄환을 의미하고, 0는 적군기에서 발사된 탄환이다. 아군기에 발사된 탄환이나 적군기에서 발사된 탄환이 상대방에 명중하지 못하면 계속 같은 방향으로 진행하다가 설정된 영역 밖으로 벗어나면 소멸되도록 하였다. 아군기는 좌, 우, 정지 동작을 하며 적군기는 좌, 우 이동만 한다. 아군기는 영역을 벗어나지 못하도록 이동중에 X 좌표가 0보다 작으면 X 값은 80으로 하고 X 값이 80보다 크면 X값은 0으로 초기화 되도록 하였다. 적군기는 랜덤하게 생성되며 지정된 영역 밖으로 벗어나면 소멸되며 소멸과 동시에 다시 생성되도록 하였다. 이 때 화면에 존재하는 적기 수는 최대 5개가 되도록 하였다.

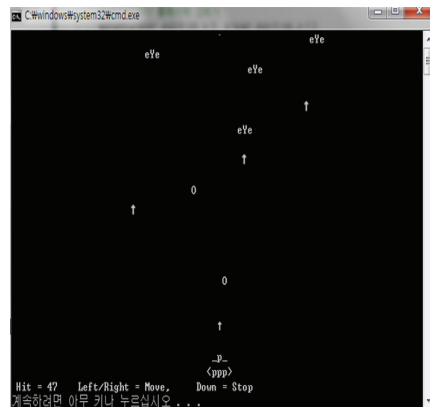


그림 4. 게임 화면
Figure 4. game monitor

성능을 평가하기 위해 적군기의 총 수는 100개로 하였다. 아군기에서 발사된 탄환이 적군기의 좌

표와 일치하면 화면 하단에 Hit 수를 증가시켰다. 게임의 성능 평가를 위해 적군기가 100개 생성동안 격추나 영역 밖으로 이동하여 화면내에서 모두 사라지게 되면 이때 까지의 Hit 수와 명중률을 카운트하였다. 적군기를 공격하는 방법으로는 플레이어가 상대방의 위치나 적군기로부터 발사되는 폭탄을 피하면서 공격하는 방법과 아군기의 이동은 플레이어가 조작하고 공격은 유전자 알고리즘을 적용하여 알고리즘에서 유리한 위치에서 공격하도록 하였다. 플레이어에 의해 탄환을 발사하도록 하면 플레이어의 게임 숙련도나 순발력에 따라 차이가 많이 나게 된다. <표 2> 게임 초보자, 중급자, 숙련자를 선정하여 각각 5번의 게임을 한 후 Hit 수를 평균한 값이다. 적군기는 랜덤하게 100개가 생성되었지만 게임 종료시까지 Hit수가 100이 안 되는 이유는 게임 중에 명중시키지 못하고 영역 밖으로 벗어난 것을 의미한다. 여기서 명중률은 게이머가 탄환 발사한 총 갯수중에서 Hit 시킨 비율이다. 아군기의 이동과 탄환 발사를 사람이 조작하지 않고 유전자 알고리즘을 적용한 결과 100개의 적군기 중에서 67개를 Hit 시켰으며 이 때 명중률은 82.7%였다. 이렇게 경력 3년의 게이머의 결과 보다도 Hit 수는 나쁘고 명중률이 좋은 것은 이유는 유전자 알고리즘 적용에 두 가지 요인(자신의 위치와 상대방 위치)을 만족하는 해를 찾는데 시간이 소요되지 않았기 때문이다.

표 2. 플레이어의 게임 결과

Table 2. result of game

실험자	Hit수/명중율	비 고
A	46 / 39.2	경력 1년 미만
B	54 / 55.9	경력 3년
C	72 / 68.4	경력 5년 이상

그러므로 유전자 알고리즘에서 해를 찾는데 걸리는 시간을 단축하기 위해 아군기의 이동은 게이머가 조작하고 탄환 발사에만 유전자 알고리즘 적용한 결과 <표 3 >이다. <표 2>에서 Hit수가 높지 않은 것은 적군기가 랜덤하게 이동하면서 아군기로부터 공격이 이루어지기 전에 화면 밖으로 사라진 결과이다. 표2에서 탄환 발사에 유전자 알고리즘을 적용하였음에도 불구하고 명중률이 100%이 아닌 것은 유전자 알고리즘에서 상대방의 이동 패턴을 충분히 학습하지 못했기 때문이다.

표 3. 플레이어의 게임 결과

Table 3. result of game

실험자	Hit수/명중율	비 고
A	62 / 96.5	경력 1년 미만
B	77 / 96.1	경력 3년
C	85 / 97.4	경력 5년 이상

그러나 이 문제를 해결하려면 세대수를 더 충분히 늘려야 하나 그렇게 할 경우 처리 시간이 오래 걸려 게임의 속도에 문제가 발생하게 된다. 따라서 인공지능을 이용한 게임에서는 100%의 최적해를 요구하는 것 보다는 처리 속도와 적절한 절충이 필요하다.

5. 결 론

본 논문에서는 슈팅 게임에서 조건문이나 decision table을 사용하지 않고 객체의 판단에 유전자 알고리즘을 적용하였다. 아군기와 적군기를 랜덤하게 생성하여 서로 무기를 발사하여 명중시키도록 하였다. 적군기는 랜덤하게 100개를 생성하였으며 이 중에서 영역 안에 밖으로 나가지 않은 것 중에서 명중시킨 것을 대상으로 하

였다. 적군기를 공격하는 방법으로는 플레이어가 상대방의 위치나 적군기로부터 발사되는 폭탄을 피하면서 공격하는 방법과 아군기의 이동은 플레이어가 조작하고 공격은 유전자 알고리즘을 적용하여 알고리즘에서 유리한 위치에서 공격하도록 하였다.

평가는 게임의 초보자, 중급자, 숙련자 세 사람에게 각각 5번을 게임하여 적군기의 Hit 수와 명중률을 평균하였다. 실험 결과 이동과 공격을 플레이어가 한 경우와 이동은 플레이어가 하고 공격은 유전자 알고리즘을 적용한 결과가 Hit수와 명중률에서 높게 나타났다. 그러나 속도를 필요로 하는 게임에서는 인공지능을 적용하는데 알고리즘 처리 속도에 제한이 있으므로 적절한 절충이 필요하다고 생각된다.

References

- [1] Penelope Sweetser, and Peta Wyeth, *A model for evaluating player enjoyment in games*, Game Developers Conference, 2004.
- [2] David M Bourg, and Glenn Seemann, *AI for game developers*. O'Reilly Media, 2004.
- [3] JSturtevant. N., and Buro. M. Pa., *Partial pathfinding using map abstraction and refinement*, In Proceedings of AAAI, pp. 47-52.
- [4] Daniel Fu, Ryan Houlette, and Stottler Henke, *Putting AI in entertainment: An AI authoring tool for simulation and games*, IEEE Intelligent and Systems, Vol. 17, No. 4, July/August 2002.
- [5] Brian Magerko, John E. Laird, Mazin Assanie, *AI character and directors for interactive computer games*, Proc. AAAI, 2004.
- [6] Tho, Dang Xuan, *Genetic algorithms and*

application in examination scheduling, Grin Verlag, 2013.

- [7] Adalberto Ramirez Muz, *Handbook of genetic algorithms: New Research*, Nova, 2012.
- [8] Vose, Michael D., *The simple genetic algorithm*, Mit Press, 2007.
- [9] Juul. J., *Working with the player's repertoire*, International Journal on Intelligent Games and Simulation. pp 54-61. 2004.
- [10] W. Sterren, *Tactical path-finding with GA, in game programming em 3*, Boston: Charles River Media, pp. 294-306. 2002.

유전자 알고리즘을 이용한 슈팅 게임 구현

이면섭

인천대학교 컴퓨터공학부

요 약

본 연구는 슈팅게임에서 플레이어가 자신의 전투기를 이동하면서 탄환을 발사하여 상대방의 전투기를 격추시키는 게임이다. 상대방 전투기는 랜덤하게 생성되며, 영역 밖으로 벗어나면 소멸된다. 상대방을 공격하는 방법은 2가지로 하였다. 첫 번째는 플레이어가 자신의 전투기를 이동하면서 공격하도록 하였다. 두 번째는 자신의 전투기 이동은 플레이어가 이동하고 공격은 유전자 알고리즘을 적용하였다. 이동은 좌, 우, 정지 세 가지 동작이 있다. 적군기는 랜덤하게 100개를 생성하였으며 영역 밖으로 나간 것은 평가에서 제외하였다. 영역 안에 있는 적군기를 격추시킨 수와 명중률을 평가하였다. 실험 결과 유전자 알고리즘을 적용한 두 번째 실험에서 좋은 결과를 얻었다. 좌, 우, 정지 세 가지 동작이 있다. 적군기는 랜덤하게 100개를 생성하였으며 이 중에서 영역 안에 밖으로 나가지 않은 것 중에서 격추시킨 것과 명중률을 평가하였다. 실험 결과 유전자 알고리즘을 적용한 두 번째 실험에서 좋은 결과를 얻었다.

감사의 글

본 논문은 인천대학교의 2013학년도 학술연구조성비를 지원 받음.



Myun Sub Lee received the B.S. degree in Electrical Engineering from Kookmin University. in 1985. He received the M.S. degree in the department of Electrical Engineering from Inha University in 1987 and the Ph.D. degree in the Department of Computer Engineering from Kookmin University in 2005. He was a professor in Department of Computer Science and Engineering at Incheon University from 1990 to 2014. His current research interests include artificial intelligence, intelligent Games.

E-mail address: nantian@incheon.ac.kr