

# FPS bot 사격 알고리즘

컴퓨터공학과

201433813

신승빈

## 1. 개요

NPC(non Player Character)를 지원하는 슈팅 게임에서는 일명 bot 이라고 하는 인공지능을 가진 플레이어 기능을 지원해준다.

이 bot 은 주로 처음 게임을 접하는 유저에게 게임에 대한 지식을 가르치기 위한 튜토리얼용으로 사용되거나, single player 게임에서의 경우 유저에게 끊임없이 압박을 주기위한 역할로 사용될때도 있고, multi player 게임에서의 경우 유저간의 대결에서 부족한 유저의 공백을 채우기 위한 역할로도 사용될 수도 있다. 주로 전자의 경우, NPC 의 능력치를 조절하여 유저와 같은 무기를 사용하더라도 더 높은 데미지를 입히거나, 유저보다 더 높은 체력을 가지게 하는 형태로 구현된다. 후자의 경우, 사용자와 비슷한 능력치를 가진 AI 들이 특정한 알고리즘에 의해 작동하면서 대부분의 유저보다 뛰어나지 못한 성능을 보여주는 정도로 구현되기도 한다.

이들 bot 은 대체로 일명 aimhack 이라 불리는 자동 조준 기능을 통해 구현되는데, 주로 타겟에 미리 지정된 부위로 aim 을 이동시키는 형태로 구현된다. 이 bot 의 성능이나 난이도가 높은 경우, 자동 조준을 하기위한 지연 시간이 짧고, 주로 높은 피해를 입힐 수 있는 특정 부위를 조준하도록 프로그래밍 해두고, 난이도가 낮을 경우, 지연 시간이 길고, 높은 피해를 입힐 수 있는 부위보다는 일반적인 피해를 입히는 부위를 조준하거나, 또는 일부는 실제로 사격 애니메이션만 보일뿐 사격하지 않거나 일부로 피해받지 않는 위치를 쏘게 하는 방법으로 그 난이도를 조절한다.

낮은 난이도로 구현된 bot 의 경우, 낮은 수준의 행동 알고리즘과 느린 반응속도로 인해 유저들은 연습용 존재 그 이상으로 생각하지도 않고, 그렇기 때문에 multi player 게임에서 실제유저가 부족해서 대체하는 bot 이 자리를 채우는 것을 매우 싫어하는 경향을 보인다.

높은 난이도로 구현된 bot 의 경우도 유저들은 마찬가지로 그에대해 우호적이지 못한 반응을 보이는데, 이는 전적으로 낮은 수준의 행동 알고리즘은 그대로이지만, 사람보다 말도안되게 빠른 반응속도를 보이고 치명적인 부위를 사격하는 알고리즘에 기인한다. 슈팅 게임에서 사람들에게 흥미를 이끄는 요소는 남보다 빠르게 적을 조준하고 사격하는 능력뿐만아니라, 남이 자신을 조준하지 못하게 만드는 무빙능력, 특정한 상황을 풀어갈 수 있도록 적절한 무기를 선택하는 능력 등을 상대방과 경쟁하는 과정에서 나오는건데, 이들 bot 은 그런 경쟁적인 부분은 존재하지 않은채 유저는 인정할 수 없는 속도로 자신을 죽이기 때문에 유저들은 이에 대해 매우 불만족스러워한다.

또한 유저들은 이런 bot 의 말도안되는 조준 능력으로 어이없어 하는 경우도 존재하는데, 유저의 눈에는 보이지 않을 정도로 먼 거리에 있는 적을 말도 안되는 부위를 맞춰서 제압하는 경우를 예로 들 수 있다. 몇 백미터 떨어진 거리에 있어서 유저는 수십발을 쏘도 한두발 맞출까 말까한 상황에서 bot 은 단지 미리 지정된 aimhack 알고리즘에 의해 첫 발에 자신의 머리를 맞춘다던가, 서너발만에 자신의 몸을 맞춰서 제압한다면 유저들은 분노를 넘어 어의없음을 느낄 수밖에 없다.

이렇듯 bot 이 유저들에게 적절한 경쟁상태를 부여하기 위해서는 여러 가지 능력이 요구되는데, 이러한 요구를 크게 나눈다면 적을 조준하는 능력, 적으로부터 피하는 능력, 상황에 맞는 무기를 선택하는 능력이라 할 수 있다.

여기서 필자는 bot 이 적을 조준하는 능력. 특히 상호간의 거리를 인식하고 이를 기반으로 사격을 할것인지, 말것인지를 결정하는 것과, 만약 사격을 한다면 어느 부위를 조준하여 사격할것인지를 결정하는 신경망을 구축해보도록 하겠다.

## 2. 구현 개념

위 문제는 크게 사격을 할지 여부를 결정하는 알고리즘과, 만약 사격을 한다면 어느 부위를 조준할 것인지를 결정하는 알고리즘으로 나눌 수 있다.

첫 번째 알고리즘의 경우 사격을 하는 Actor가 넓이 Area 만큼인 표적  $T_{Area}$  를 X 만큼 떨어진 거리에서 사격할 때 명중률을 R 이라고 가정하자. 이 때 임의의 거리 x 만큼 떨어진 표적 T의 넓이 area가

$$T_{area} = W_x * W_y$$

일 때, 사격여부 함수

$$BooleanFire(x, T_{area})$$

를 구하는 문제라 볼 수 있다. 일반적으로 유저가 먼 거리에 있는 타겟을 사격할지 말지 여부는 1. 맞출 수 있느냐, 2. 제압할 수 있느냐를 기반으로 결정할 것이다. 이는 결국 사격을 하는 것이 효율적인지 여부를 결정하는 것이라 할 수 있다. 이 때 이 효율 efficiency를 특정한 발사 횟수에서의 명중 횟수라 가정한다.

$$efficiency = hit / fireCount$$

최종 출력 이전의 값을 나타내는 fire()함수는 efficiency 보다 높아야하고, 효율 efficiency는 결국 Fire()함수의 마지막 출력여부를 결정하는 일종의 문턱값이라고 볼 수 있다.

$$Fire(x, T_{area}) = \begin{cases} \text{if } fire(x, T_{area}) \geq efficiency : \text{return true} \\ \text{if } fire(x, T_{area}) < efficiency : \text{return false} \end{cases}$$

첫 번째 알고리즘은 분류 문제로 바뀌게 되고, 우리는 분류 학습을 시키기 위한 지도 값인 정답 값이 원-핫 인코딩한 상태로 필요하다. 이 값을 correct라 한다면,

$$correct = \begin{cases} fireCountCombinationhit * (R)^{hit} * (1-R)^{(fireCount-hit)} & \text{if } correct \geq efficiency \\ & \text{if } correct < efficiency \end{cases}$$

그런데, 이때의 hitRate는 표적 T의 넓이  $T_{area}$ 에 의해 결정된다. 표준적인 상황  $T_{Area}$ 를 명중할 확률이 R일 때 사격자는  $T_{Area}$ 를 균등한 비율로 맞춘다고 가정한다면, R은  $T_{Area}$ 의 넓이에 비례하게 결정된다. 이 때, 100% 맞을 확률을 가지는  $T_{100}$ 를 구할 수 있다.

$$T_{100} = \frac{T_{Area} * 1}{R}$$

타겟의 넓이  $T_{area}$ 가 이미 주어졌고, 타겟과의 거리와 타겟의 넓이 사이의 관계를 통해 타겟과의 거리에 적합하게 보이는 타겟의 넓이  $T_{area}'$ 를 구한다면,  $T_{area}'$ 와  $T_{100}$ 의 비율은  $T_{area}'$ 에서의 명중률이라 할 수 있다. 이 때의 명중률을 R'이라 한다.

위 식들을 정리하여 사격 여부를 결정하는 네트워크의 학습 알고리즘을 작성한다.

만약 사격을 하기로 결정했다면, 어느 부위를 사격할지에 대한 결정을 내려야한다. FPS 게임에서는 높은 피해를 통해 빠르게 제압할 수 있지만 명중시키기 어려운 머리 부위와, 낮은 피해를 많이 입혀야 하지만 피격 부위가 넓어 맞추기 쉬운 몸통 부위로 나누어 피해판정을 내린다.

성인 남성의 평균 키 175cm, 어깨 넓이를 간단하게 50cm라 가정할 때, 이 표적은  $0.875m^2$ , 간단하게  $1m^2$ 라 가정할 수 있다. 또한 사람의 평균적인 등신대를 7등신으로 가정한다면,  $0.14m^2$  정도가 된다. 사람의 평균적인 머리의 크기를 어깨의 약 절반정도로 가정한다면 약

$0.7m^2$  , 계산을 간단히 하기 위해 약  $0.1m^2$  라 가정하면, 정면에서 바라본 사람의 머리와 몸의 넓이 비율은 약 1 : 8 이라 할 수 있다(남은 1 은 머리와 어깨 사이의 넓이로 계산한다).

위의 계산에서 이미 지정된 위치에서의 총 명중률  $R'$ 을 구해냈으니 머리를 맞출 명중률은

$$R_{head} = R' * 0.1$$

이고, 몸을 맞출 명중률은

$$R_{body} = R' * 0.8$$

이다.

bot은 최대한 빨리 상대를 제압해야한다. 가장 빨리 제압하는 방법은 첫 사격을 머리에 맞추는 것이지만, 명중률에 따라서 맞을수도 있고, 빗맞을 수도 있다. bot이 사격을 할 위치를 결정하는 것은 두 부위 간의 명중률 비교에 의해 결정된다.

머리를 맞췄을 때 피해량을 100 이라 가정하고, 몸을 맞췄을 때 피해량을 20 이라 가정한다면, bot은 총 fireCount 번 사격을 했을 때 1 번 이상 머리를 맞출 확률과 몸을 5 번 이상 맞출 확률을 비교하여 결정할 수 있어야하고, 이 값을 원-핫 인코딩하게 정답 데이터로 지정해주면 된다.

$$hitProbability_{head} = \binom{R_{head}}{1} (1 - R_{head})^{fireCount - 1}$$

$$hitProbability_{body} = \binom{R_{body}}{5} (1 - R_{body})^{fireCount - 5}$$

$$one_hot_encoding = \begin{cases} 1 & \text{if } hitProbability_{head} \geq hitProbability_{body} \\ 0 & \text{if } hitProbability_{head} < hitProbability_{body} \end{cases}$$