



Inteligencia Artificial

Trabajo 1

Integrantes

Renzo Eduardo Damian Gomez

U201715758

Luis Enrique Guadalupe Carrillo

U201716383

Sección

SI62

Profesor

Wester Edison Zela Moraya

Ciclo Académico

2020-01

Índice

I. Planteamiento del problema

II. Fundamentación

III. Explicación heurística

IV. Detalles

V. Pruebas de uso

VI. Referencias

I. Planteamiento del juego

Gunbound es un videojuego multijugador en el cual 2 o más adversarios conduciendo vehículos de guerra se enfrentan en distintos escenarios en los cuales tienen que derrotar al resto destruyendolos lanzando misiles hasta dañar por completo el vehículo enemigo. Este popular juego del 2003 se caracterizó porque a pesar de ser los mismos escenarios y los mismos protagonistas, las partidas nunca eran iguales, siempre variaba pues, a pesar de ser un jugador experto y dominar todos los mapas, era casi imposible siempre saber la mira correcta (combinación entre la potencia y el ángulo del disparo).

Por esta razón, Gunbound 2.0, el proyecto a desarrollar en este trabajo, busca crear al bot capaz de realizar el rol del “Player perfecto”, que, en cualquier circunstancia, este sea capaz de atinar siempre al enemigo. Por lo que se adaptarán ciertas técnicas para que se pueda realizarlo.

Finalmente, en este “Trabajo 1” se resolverá un problema en particular, que en un escenario con una sola plataforma plana, la computadora (bot) sea capaz de lanzar el proyectil “perfecto” con el uso de un algoritmo genético que se explicará más adelante tanto su implementación como la heurística que permitirá resolver el problema en cuestión.

II. Fundamentación

Como se mencionó anteriormente, se busca adaptar un algoritmo de Inteligencia Artificial al juego en sí, por lo que se analizaron algunos quedando el Algoritmo Genético como el más apropiado para este trabajo.

Se optó por este algoritmo ya que lo que se busca es que una solución a un problema en específico utilizando un método optimizado. Además con la ayuda de una Función de Idoneidad (Fitness Function) se encargará de evaluar qué tan buenas son las posibles soluciones.

Como menciona Gestal (2010) “Para alcanzar la solución a un problema se parte de un conjunto inicial de individuos, llamado población, generado de manera aleatoria. Cada uno de estos individuos representa una posible solución al problema y estos individuos evolucionarán.”

Por último, las soluciones más adaptables (superiores) tienen mayor probabilidad de heredar su contenido a las generaciones futuras.

La aplicación de los datos ingresados en el input serán de manera aleatoria pero conservarán esta estructura.

- Población inicial: 10 cromosomas

- Número de dígitos de cromosomas en binario: 11

- El cromosoma al estar en sistema decimal será la unión de la potencia y ángulo del disparo teniendo como máximo de potencia 64 y ángulo 32 en el sistema decimal.

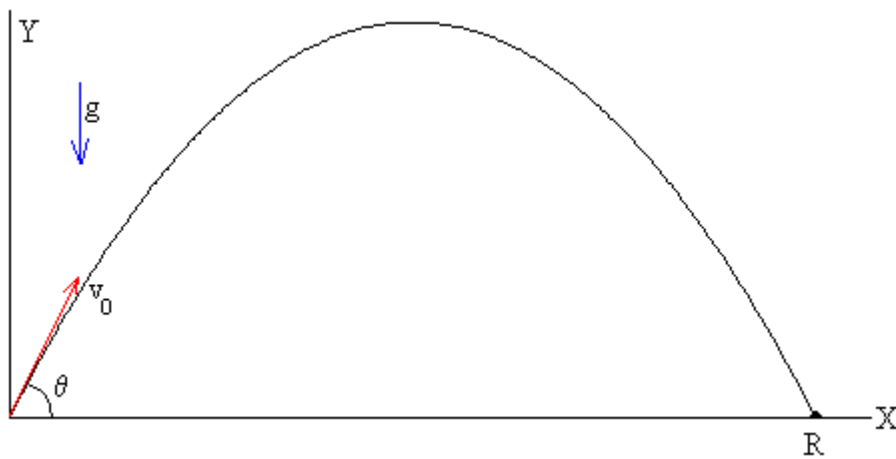
Ejemplo:

“potencia” “ángulo”

6	4	3	2
4	2	2	0
3	5	2	2
3	4	2	0

- La función fitness se calculará de la siguiente manera:

Primero: La trayectoria de la bala será un movimiento parabólico simple. Por lo que la fórmula entre el ángulo y la potencia nos dará una simulación de cómo caería el proyectil.



Segundo: La siguiente fórmula física nos dará la distancia (x) en donde caerá el proyectil

$$X_{\text{máx}} = \frac{v_i^2 * \sin 2 \theta}{g}$$

Tercero: Teniendo en cuenta estas condiciones, utilizando Optimization of a Function, el valor de “x” vendría a ser la distancia donde cae el proyectil, y el valor de la función de (x) sería

$$f(x) = 700 - x$$

Debido a esta función de idoneidad, mientras más se acerque a 0, la solución será más óptima.

De igual forma, al hacer el cruce de cromosomas se compararon 2 cromosomas de manera aleatoria y se escogió al de mayor fitness.

III. Heurística

Teniendo en cuenta que el estado inicial nos mandará 10 cromosomas creados aleatoriamente con 11 dígitos en el sistema binario con un máximo valor de 11111111111

POBLACIÓN INICIAL	FUNCIÓN DE IDENTIDAD
X_1	$f(X_1)$
X_2	$f(X_2)$
X_3	$f(X_3)$
X_4	$f(X_4)$
...	

La función heurística recibe de parámetro el score el cual retorna la diferencia que falta o le sobró recorrer al proyectil para que logre llegar al objetivo.

```
class chromosome:
    def __init__(power, angle):
        self.xmax = 0
        self.score = 0
        self.power = power
        self.angle = angle

    def get_score(xmax):
        score = abs(700-xmax)
        return score
```

El porqué de la selección de esta heurística se debió a que se hallaría un valor comparable en el cual cada cromosoma al ser evaluado y finalmente el programa pueda escoger los más exitosos teniendo como meta al score.

IV. Detalles

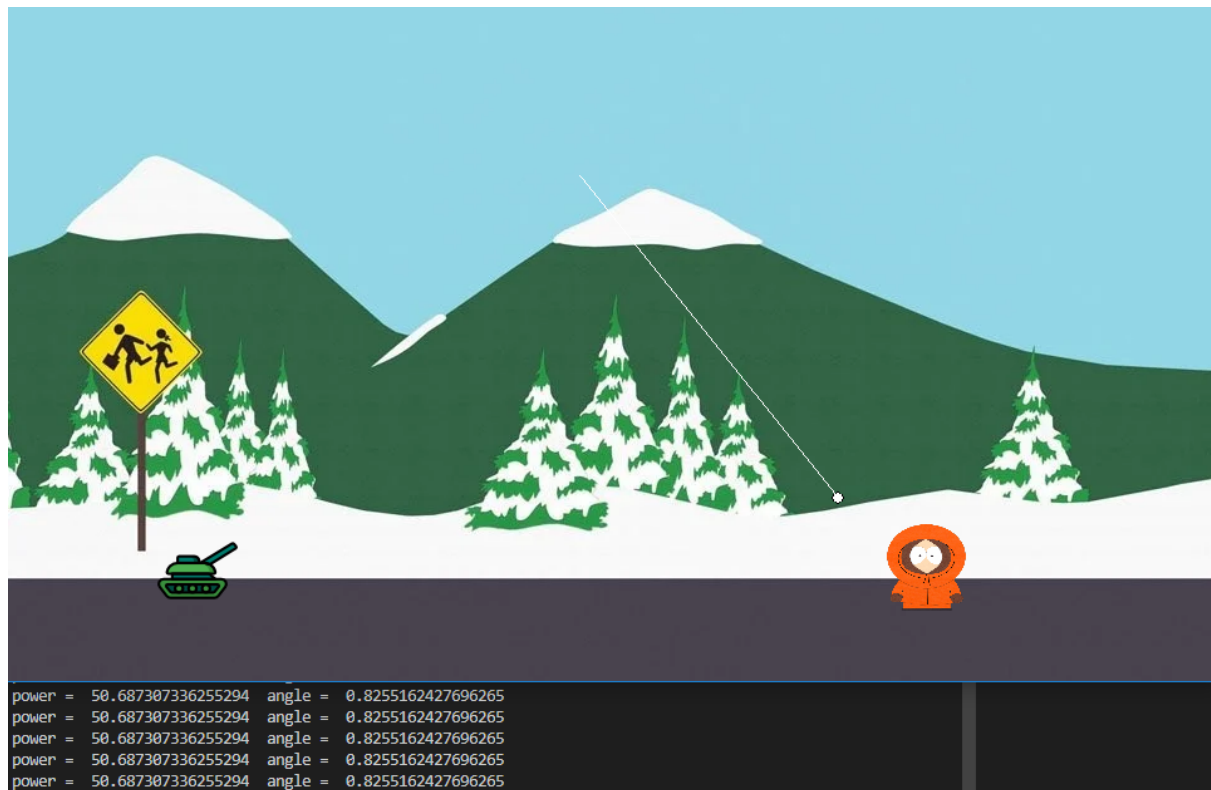
La única librería que se usó en el trabajo es Pygame (librería estándar de Python diseñada para juegos) por la posibilidad que nos brindó de dibujar la “puntería” y “bala” del tanque para calcular y realizar el tiro.

V. Pruebas

Ejemplo 1 de potencia y ángulo del disparo.



Ejemplo 2 de posibles disparos certeros



Ejemplo de la ejecución de los cromosomas

```
pygame 2.0.0.dev6 (SDL 2.0.10, python 3.8.5)
Hello from the pygame community. https://www.pygame.org/contribute.html
power = 1 angle = 0.4416094210009814
184
0
power = 54 angle = 0.04906771344455348
370
0
power = 37 angle = 0.9322865554465163
463
0
power = 31 angle = 0.6869479882237488
397
0
power = 30 angle = 1.2757605495583906
289
0
power = 22 angle = 0.14720314033366047
259
0
power = 16 angle = 0.19627085377821393
237
0
power = 38 angle = 0.4416094210009814
+ Parabolica.py(main) x □
```


VI. Referencias

-Marcos Gestal. (2010). Introducción a los Algoritmos Genéticos y la Programación Genética. Recuperado el 13 de septiembre del 2020, de Universidade da Coruña en:

https://d1wqtxts1xzle7.cloudfront.net/55091016/algoritmos-geneticos-libro-bueno.pdf?1511442140=&response-content-disposition=inline%3B+filename%3DIntroduccion_a_los_Algoritmos_Geneticos.pdf&Expires=1600147341&Signature=D3mIaLX7I3kqLxah~3P1LBJPmpK2dIeCPkxf~GeE9DHX5zonaH5aH75todXkkRNVzC9Be7m5HdTiyWw8UwbGiqQom0D-TAxmWtFVQeA1YiUvHNYXbElwFyXXNUi6DIX-fBJ-oJbjLNvgun0iuzY6fHJQ5J58xnOWBpVB5Dh90iD9YpN9T75u2dcAhsAvz2dC4ltGizb8Zws5GagoXauOWoDhHXjh8ksPKzwuJdaD5H--4xBQn7agWIUh~nUHBpaW74b0nBAHWiG-9kcyoDGNyZYo23eUQeFxqyKW8Xu6LLYwQHt77TvgvbARkxIlzzSHKGI2eNTDl9JZPLd7dQmfPg__&Key-Pair-Id=APKAJLOHF5GGSLRBV4ZA

-