

Primero se describen las ecuaciones de movimiento del sistema:

Cañon ofensivo:

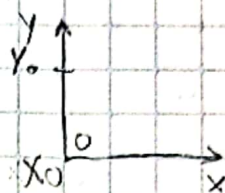
$$V_x = V_0 \cos(\alpha t)$$

$$V_y = V_0 \sin(\alpha t) - gt$$

$$x_0 = V_x t$$

$$y_0 = V_0 + V_y t - \frac{1}{2}gt^2$$

Sistema referencia cañon ofensivo



Según el sistema de referencia el cañon ofensivo se encuentra en la posición $(0,0)$, por tanto se omite la posición inicial en x .

Cañon defensivo:

$$V_x = V_0 \cos((\alpha + \pi/2)t)$$

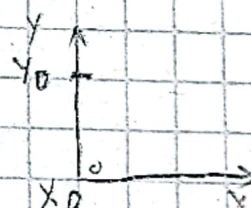
$$V_y = V_0 \sin((\alpha + \pi/2)t) - gt$$

$$x_D = V_x t$$

$$y_D = V_0 + V_y t - \frac{1}{2}gt^2$$

Sistema de referencia

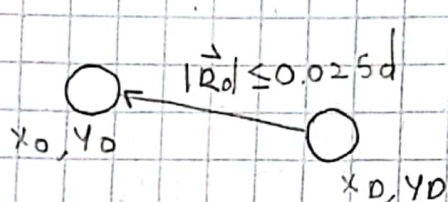
Cañon defensivo



Al agregarle la fase a la velocidad, implica que el sentido de la bala es hacia la izquierda. Por facilidad en los cálculos, se toma otro sistema de referencia para el cañon defensivo, por lo que su posición inicial en x , también será cero.

Ecuaciones de efectividad.

1. El disparo defensivo sea efectivo si y solo si

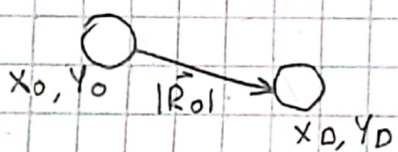


$$\vec{R}_D = (x_o - x_D)\hat{i} + (y_o - y_D)\hat{j}$$

Es decir la magnitud del vector \vec{R}_D sea menor a $0.025d$

$$\sqrt{(x_o - x_D)^2 + (y_o - y_D)^2} \leq 0.025d$$

2. El disparo ofensivo sea efectivo si y solo si



$$|\vec{R}_o| \leq 0.05d$$

$$\vec{R}_o = (d - x_o)\hat{i} + (H_D - y_o)\hat{j}$$

$$\sqrt{(d - x_o)^2 + (H_D - y_o)^2} \leq 0.05d$$

Planteamiento solución

Inicialmente se planea dos clases, una para el cañon defensivo y otra para el cañon ofensivo, donde cada una de ellas tendrá sus atributos de condiciones iniciales y sus métodos, como se describe a continuación.

Clase cañon ofensivo:

Atributos: Float d → separación entre cañones
Float Xo=0 → posición in X del cañon ofensivo
Float Yo → posición in y del cañon defensivo
Float Ro=0.05*d → efectividad de destrucción

Métodos: void disparosOfensivos : se encarga de generar 3 disparos ofensivos que comprometa la integridad del defensivo. Los parámetros de entrada que tendría sería las coordenadas del cañon defensivo, la velocidad inicial, dada por el usuario.

- Además se tendrán los método set y get, para cada uno de los atributos, ya que estos serán privados y se irán modificando por el usuario.

Clase cañon defensivo:

Atributos: Float d → separación entre cañones
Float Xn → posición in X del cañon defensivo
Float Yn → posición inicial Y del cañon defensivo
Float Rd=0.025*d

Métodos: void disparosDefensivos : se encarga de generar 3 disparos ofensivos que comprometan la integridad del cañon ofensivo. Los parámetros de entrada que tendría sería las coordenadas del

- el cañon ofensivo y la velocidad inicial, dado por el usuario.
• void disparoDefensa → puede generar 3 disparos defensivos teniendo en cuenta el disparo ofensivo, sin importar la integridad del cañon ofensivo. los parámetros de entrada que se tendrían son los objetos: ofensivo y velocidad del defen. inicial dado por usuario y también la velocidad inicial con la que se genera el disparo ofensivo y el respectivo ángulo.
- void disparoSinAfectacion → puede generar 3 disparos defensivos pero tiene que estar verificando que no afecte la integridad de ambos cañones. los parámetros de entrada serían los coordenados del cañon ofensivo, velocidad inicial con la que se generan el disparo defensivo y los parámetros de velocidad inicial y ángulo con la que se genera el disparo ofensivo.
- Además se realizan los métodos set y get para cada uno de los atributos.