

ESCUELA POLITÉCNICA NACIONAL

FACULTAD DE INGENIERÍA EN SISTEMAS

SOBREVUELO

Correa Arango Adrian Fransico

Francisco.correa01@epn.edu.ec

Quishpe Cabezas Stiv Stalyn

stiv.quishpe@epn.edu.ec

Oñate Ian Andres

Yanez Sandoval Jorge Luis

Jorge.yanez01@epn.edu.ec

PROFESOR: Jonathan Zea

jonathan.zea@epn.edu.ec

29 de noviembre, 2024

ÍNDICE

1.	OBJETIVOS.....	3
2.	INTRODUCCIÓN.....	3
3.	DESCRIPCIÓN DEL CÓDIGO.....	3
4.	DESAFÍOS ENCONTRADOS	5
	4.1. Selección de la Fórmula Adecuada.....	5
	4.2. Aprendizaje de Librerías	5
	4.3. Integración de Componentes	5
5.	RESULTADOS	5
6.	CONCLUSIONES.....	6
7.	REFERENCIAS	7

1. OBJETIVOS

- Proponer las ecuaciones que definen la trayectoria del proyectil en función de las condiciones iniciales, como la velocidad, la altura del avión y la distancia al objetivo.
- Ilustrar de manera animada la trayectoria de la bomba y destacar el punto de impacto, ya sea dentro o fuera del cráter, para facilitar el análisis visual del fenómeno.
- Analizar como el modificar las condiciones iniciales puede afectar a la trayectoria de la bomba y el punto donde impacta.

2. INTRODUCCIÓN

El presente trabajo, desarrolla una simulación la cual tiene como fin dibujar la trayectoria que sigue una bomba lanzada desde un avión desde una cierta altura hacia el piso. La bomba debe caer dentro de un cráter que toma la forma de un trapecio isósceles invertido. Para la solución que se desarrollará, se combinan conceptos tanto de física, modelado matemático y programación con el fin de analizar el movimiento del proyectil (bomba), tomando como parámetros condiciones iniciales capaces de cambiar el resultado de la simulación.

El objetivo principal del presente trabajo consiste en visualizar el comportamiento de la bomba desde el momento en el que sale del avión hasta el momento del impacto, considerando variables como la altura inicial del avión, la velocidad con la que parte la bomba, el ángulo de las paredes del cráter y la distancia horizontal que existe desde el avión al inicio del cráter al principio de la simulación. Esta simulación fue implementada en Python con herramientas como Tkinter y Matplotlib, permitiendo ajustar los parámetros por medio de una interfaz gráfica y el análisis de diferentes escenarios.

3. DESCRIPCIÓN DEL CÓDIGO

Para que el programa sea capaz de dibujar la trayectoria de la bomba, debemos saber inicialmente que la bomba va a describir una trayectoria parabólica, por lo cual el código para dibujar la trayectoria se va a basar en las siguientes ecuaciones.

$$t = (x - x_0)/v_{(x_0)}$$

$$y = y_0 - 1/2gt^2$$

El código, para funcionar, necesita de varias librerías, entre las que destacan **numpy** para realizar cálculos matemáticos, **matplotlib** para generar gráficos, y **tkinter** para construir la interfaz gráfica interactiva.

A continuación, se define la función **trayectoria** que calcula la altura de la bomba en función de su posición horizontal y las condiciones iniciales de lanzamiento, considerando la aceleración que es la de la gravedad.

```
# Constante de gravedad
g = 9.81

# Función de la trayectoria de la bomba
def trayectoria(x, x0, y0, v0x, v0y):
    t = (x - x0) / v0x
    return y0 - 0.5 * g * t**2
```

Figura 3.1 Función para calcular la trayectoria.

Para construir la interfaz gráfica se usa la librería **Tkinter**, con la que se permitirá introducir parámetros como son:

- Altura inicial del avión (ha).
- Altura del fondo del cráter (hc).
- Longitud del cráter (L).
- Ángulo de las paredes del cráter (alpha).
- Distancia inicial del avión al borde del cráter (d).
- Velocidad inicial de la bomba (v0).
- Posición del cráter (pos_crater).

La simulación muestra el movimiento horizontal del avión y la trayectoria que sigue la bomba, misma que se calcula y grafica dinámicamente, para luego evaluar las posibles colisiones con las paredes del cráter o del suelo.

La librería Matplotlib es usada para dibujar el avión en movimiento, el cráter y su geometría, la trayectoria de la bomba y el punto de impacto.

El código verifica donde está el punto de impacto, es decir si la bomba impacta con las paredes del cráter, el suelo del cráter o el suelo fuera del cráter. Una vez que la bomba impacta uno de estos suelos se detiene la simulación.

4. DESAFÍOS ENCONTRADOS

4.1. Selección de la Fórmula Adecuada

El primero de los desafíos fue la seleccionar de la fórmula correcta para calcular la trayectoria de la bomba. Investigamos algunas ecuaciones de movimiento parabólico y finalmente optamos por una que considerara solo la gravedad y la velocidad del avión, ya que se adaptaba.

4.2. Aprendizaje de Librerías

El uso de librerías como Matplotlib y Tkinter fue un desafío al inicio ya que tienen muchas funciones útiles y se tiene que identificar cual usar y en que contexto hacerlo. Matplotlib fue esencial para la visualización gráfica de la simulación, mientras que Tkinter nos permitió crear la interfaz de usuario interactiva. Aprender a utilizar estas herramientas de manera efectiva requirió tiempo y práctica, así como la consulta de tutoriales.

4.3. Integración de Componentes

La Integración de los diferentes componentes del código, como la interfaz gráfica de usuario y la simulación, fue otro desafío importante. Ya que todos los elementos tienen que funcionar juntos sin problemas y que la interfaz tiene que ser intuitiva para el usuario final todo esto requirió varias iteraciones y ajustes.

5. RESULTADOS

La simulación muestra la trayectoria parabólica que sigue la bomba desde el momento en que es lanzada por el avión hasta que impacta en el cráter. La bomba colisiona con el cráter o con el suelo, según los datos que el usuario ingrese, lo que detiene su movimiento y finaliza la simulación.

Simulación de Caída de Bomba

Altura inicial del avión (h_a)	300
Altura del fondo del cráter (h_c)	100
Longitud horizontal del cráter (L)	1500
Ángulo de las paredes del cráter (α)	45
Distancia inicial del avión desde el borde del cráter (d)	100
Velocidad inicial de la bomba (v_0)	300
Posición del cráter (pos_crater)	1500

Iniciar Simulación

Figura 5.1. Interfaz gráfica de usuario para el ingreso de parámetro

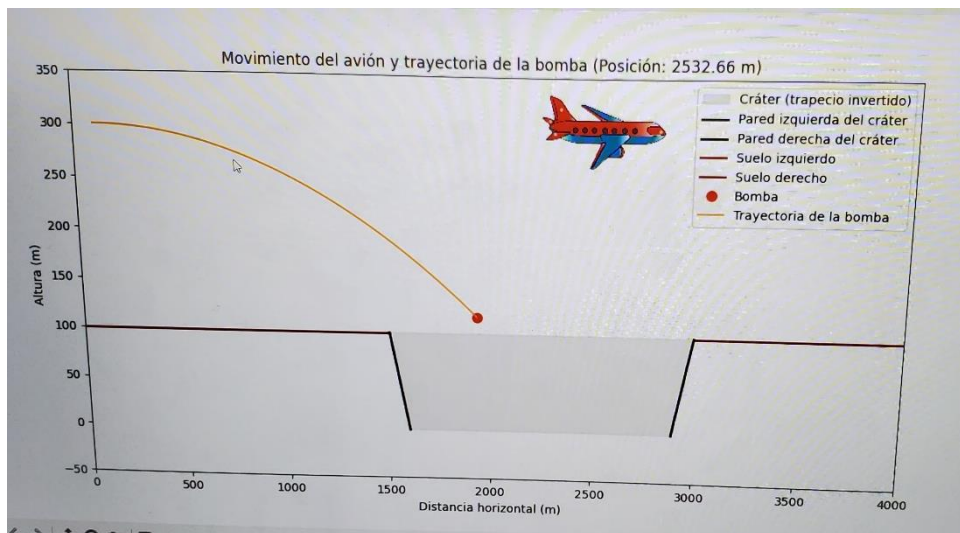


Figura 5.2. Trayectoria de la bomba con los parámetros ingresados

6. CONCLUSIONES

Este proyecto nos permitió aplicar conceptos teóricos de física y matemáticas en un contexto práctico usando código, lo que resultó en una comprensión más profunda de estos principios. La correcta selección y uso de las ecuaciones adecuadas para modelar

la trayectoria parabólica de la bomba, considerando factores como la gravedad y la velocidad inicial del avión, fue fundamental para el éxito de la simulación.

A lo largo del desarrollo del proyecto, enfrentamos y superamos varios desafíos, desde la selección de fórmulas hasta el aprendizaje de nuevas librerías y la integración de componentes. La colaboración efectiva y la comunicación constante entre los miembros del equipo fueron cruciales para resolver estos problemas y asegurar el éxito del proyecto.

7. REFERENCIAS

tkinter — Python interface to Tcl/Tk. (s/f). Python documentation. Recuperado el 30 de noviembre de 2024, de <https://docs.python.org/es/3/library/tkinter.html>

Sumich, A., & Melnikov, E. (2024, abril 12). *Biblioteca NumPy en Python*. Diveintopython.org. <https://diveintopython.org/es/learn/data-science/numpy>

Guía definitiva para dominar Matplotlib. (2022, febrero 17). Platzi. <https://platzi.com/blog/matplotlib/>