Sistemas Continuos

Ejercicio 6

Alumnos:

Buil, Delfina 43062674
Buzzacchi, Victoria 43475447

Problema:

Suponga que se desea modelar el movimiento vertical de una pelota que cae bajo la acción de la gravedad y que rebota al llegar al suelo. La aceleración gravitatoria se considera constante e igual a $g = 9,8m/s^2$.

Suponer rozamiento lineal, y con velocidad $v(t)$ positiva hacia arriba.

El sistema de ecuaciones que describe el comportamiento de la pelota es el siguiente:

$$\dot{x}(t) = v(t)$$

$$\dot{v}(t) = \begin{cases} -\frac{b_a}{m} \cdot v(t) - g & \text{si} x(t) > 0 \\ -\frac{k}{m} \cdot x(t) - \frac{b}{m} \cdot v(t) - g & \text{si} x(t) \leq 0 \end{cases}$$

Considerando los parámetros $b_a = 0,1$, $m = 1$, $b = 30$, $g = 9,8$ y $k = 100000$, calcular y graficar la posición y velocidad de la pelota hasta que la misma quede en estado de reposo.

Considerar lo siguiente:

- paso de integración $h = 0.0001$,
- detener la simulación cuando la altura sea menor a 0.00001 o un tiempo maximo de 100 unidades de tiempo.
- el valor inicial de la velocidad es 0 (cero). A modos de testeo iniciar con una altura 10.

Modelado del problema: (método Euler)

Las variables de interés en este sistema son la posición $x(t)$ y la velocidad $v(t)$). Estas describen el estado del sistema en cada instante de tiempo y son las que se necesitan calcular y analizar para entender el comportamiento de la pelota.

Condiciones iniciales:

- Velocidad inicial: $v(0) = 0.$
- Altura inicial**:** $x(0) = 10.$

Condiciones de parada de la simulación:

- Cuando la altura $x(t) < 0.00001.$
- El tiempo de la simulación sea igual a 100 unidades de tiempo.

Para resolver este problema se hace uso del método Euler, como se muestra a continuación:

- Cálculo de la altura:

$$x(t + h) = x(t) + h \frac{dx(t)}{dt}$$

$$x(t + h) = x(t) + 0.0001\, v(t)$$

- Cálculo de la velocidad:

$v(t + h) = v(t) + h\frac{dv(t)}{dt}$

$$v(t + h) = v(t) + 0.0001 - \frac{0.1}{1} * v(t) - 9.8 \qquad if\ x(t) > 0$$

$$v(t + h) = v(t) + 0.0001 - \frac{100000}{1} * x(t) - \frac{30}{1}v(t) - 9.8 \qquad if\ x(t) \le 0$$

Para resolver dicho problema de modelar el movimiento vertical de una pelota que cae bajo la acción de la gravedad y que rebota al llegar al, se llevó a cabo la siguiente implementación utilizando el método Euler:

```
class EulerMethod:

    def __init__(self, h):
        self.h = h

    def compute_next_height(self, prev_height, prev_velocity, ba, m , b, g, k):
        result = prev_height + self.h * prev_velocity
        return result

    def compute_next_velocity(self, prev_height, prev_velocity, ba, m , b, g, k):
         result = prev_velocity + self.h * self.get_acceleration(prev_height, prev_velocity, 
ba, m , b, g, k)
        return result

    def get_acceleration(self, prev_height, prev_velocity, ba, m , b, g, k):
        if prev_height > 0:
            return -ba / m * prev_velocity - g
        else:
            return -k / m * prev_height - b / m * prev_velocity - g
```

Una vez implementado, se llevaron a cabo una serie de experimentos en donde ciertos parámetros del modelo fueron cambiados para analizar qué diferencias se observan en la performance de la pelota rebotando. Todos estos se ejecutaron con el mismo paso de integración $h = 0.0001$. Los resultados se presentan a continuación:

Experimentos:

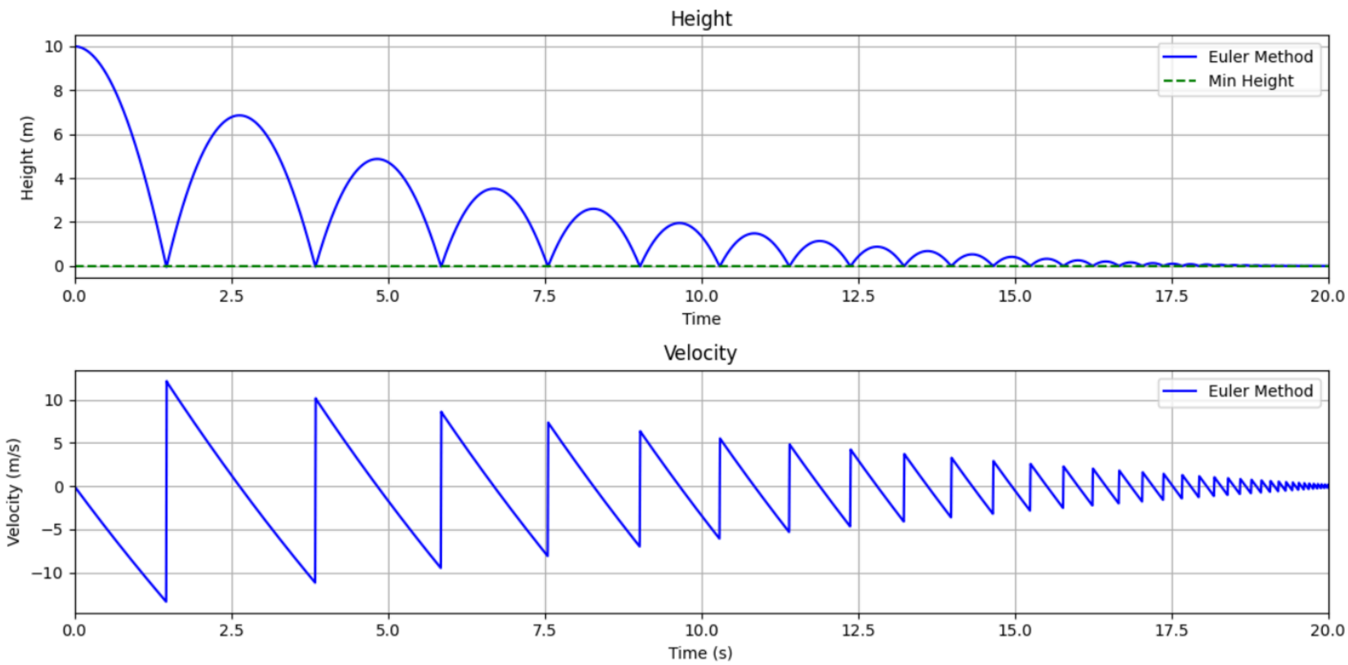
Con la herramienta `matplotlib`, la cual es una biblioteca de Python que se utilizó para crear la visualización de los datos, se generó el archivo `'simulation_plot.png'` con los gráficos correspondientes al ejecutar la simulación.

1. **Primer experimento:** variación de la masa de la pelota.

Se corrió la simulación con los parámetros por defecto:

$v(0) = 0,\ x(0) = 10,\ ba = 0.1,\ m = 1,\ b = 30,\ g = 9.8\ y\ k = 10000.$

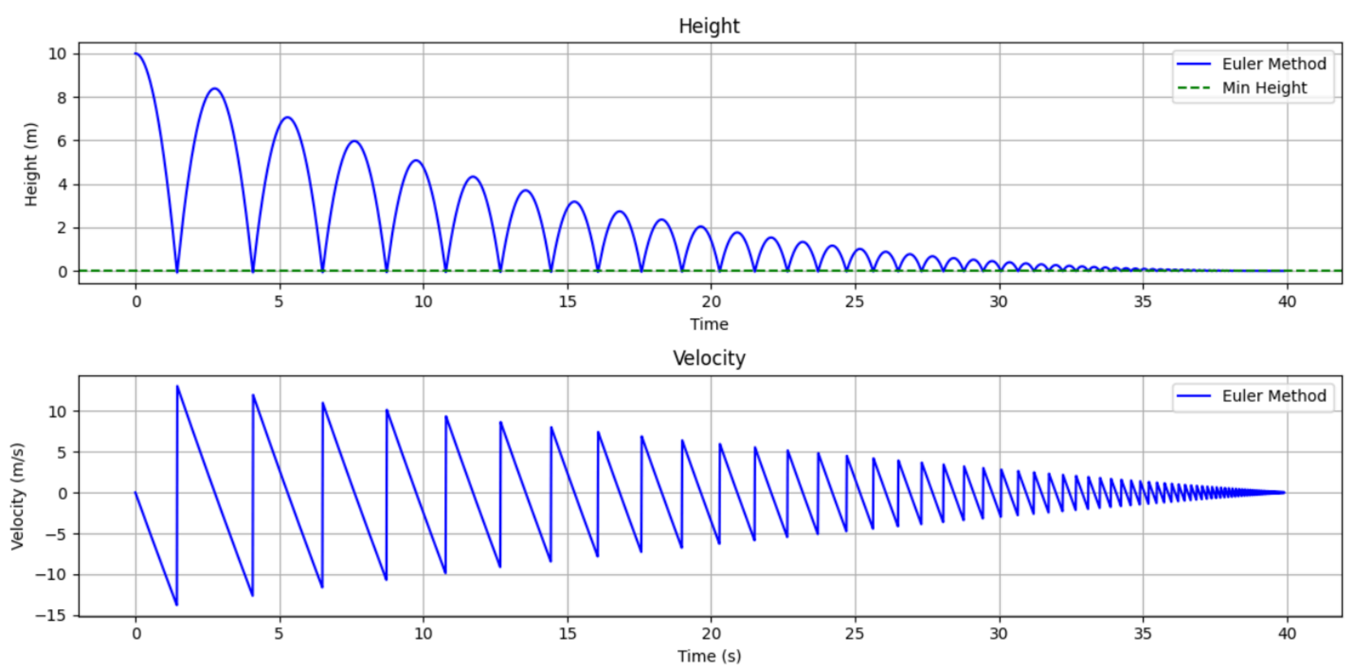
Obteniendo los siguientes resultados:



En base a estos, se modificó la masa de la pelota para observar cómo esta afecta el rebote y la velocidad.

1.2 Aumento de la masa de la pelota:

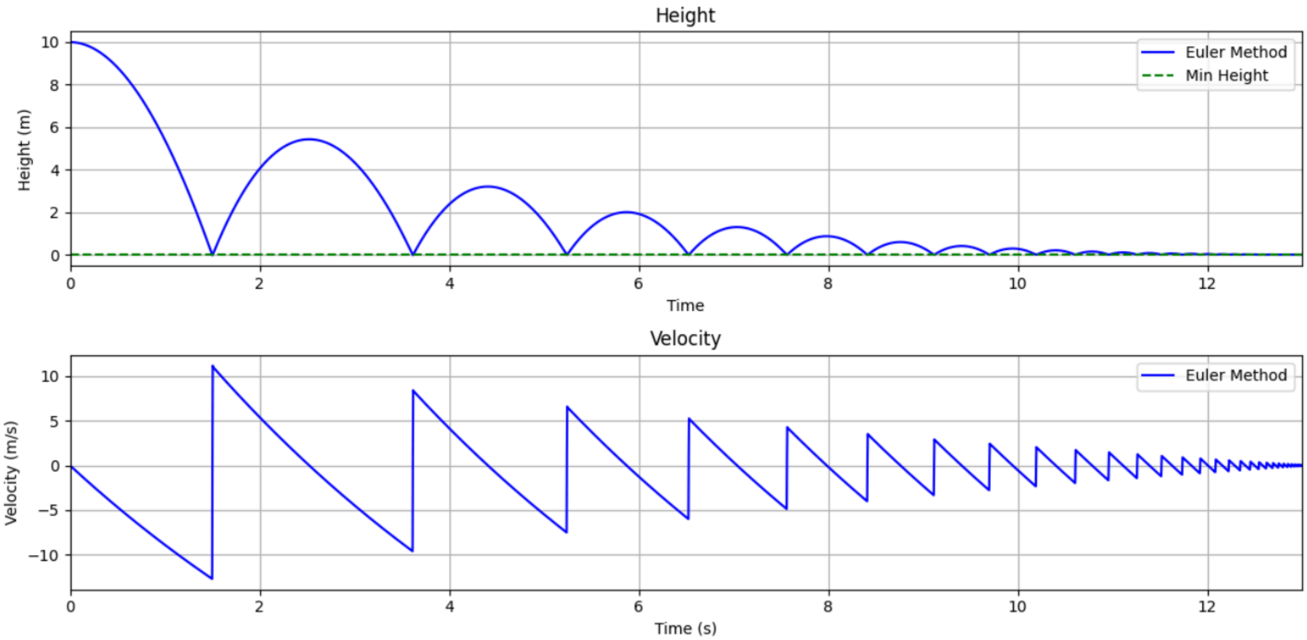
Se eligió una masa $m = 3$ y los demás parámetros se mantuvieron exactamente iguales a que el experimento inicial.



Al haber aumentado la masa de la pelota se puede observar que esta rebota una mayor cantidad de veces y por una mayor cantidad de tiempo.

1.3 Disminución de la masa de la pelota:

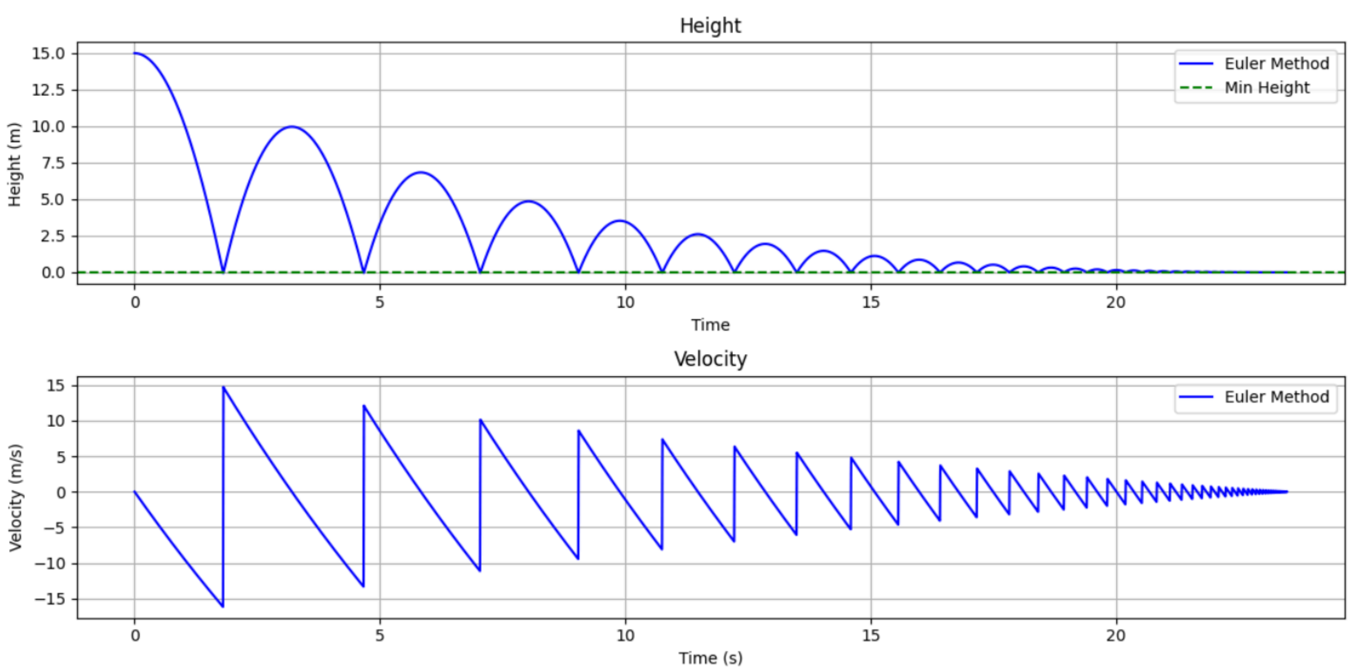
En este caso, se disminuyó la masa de la pelota por $m = 0.5$, también manteniendo los demás parámetros por defecto.



Se puede notar en este caso cómo la pelota rebota mucho menos y alcanza su estado de reposo en un menor tiempo.

2. Segundo experimento: variación de la altura inicial de la pelota.

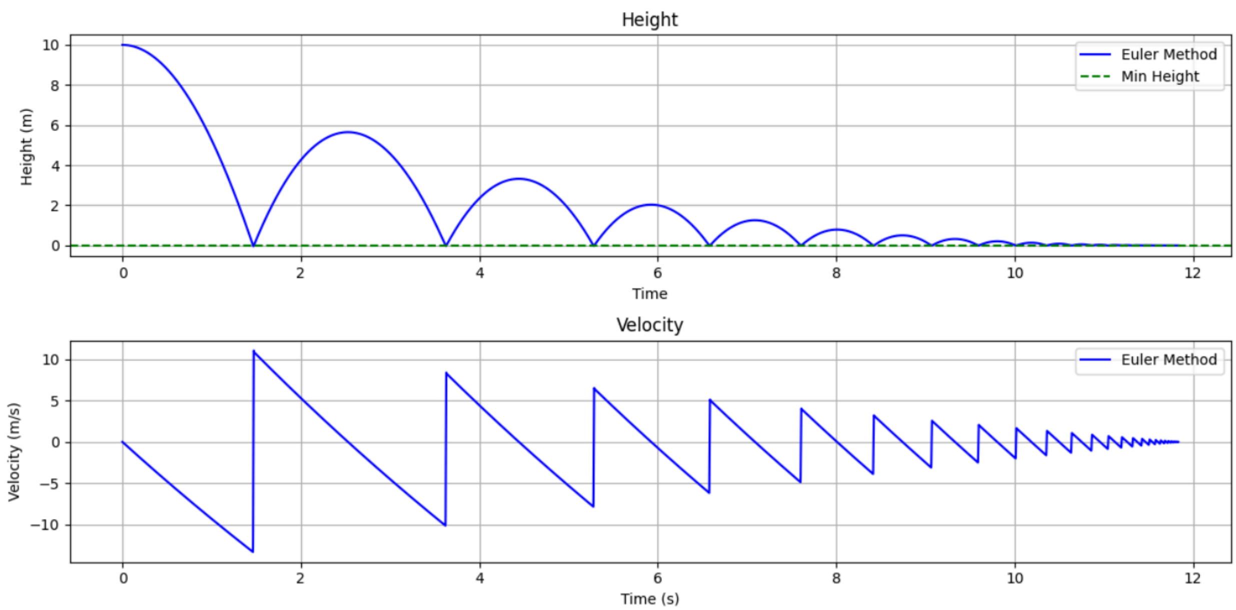
Se tomó una altura inicial $x(0) = 15$, con el resto de los parámetros por defecto.



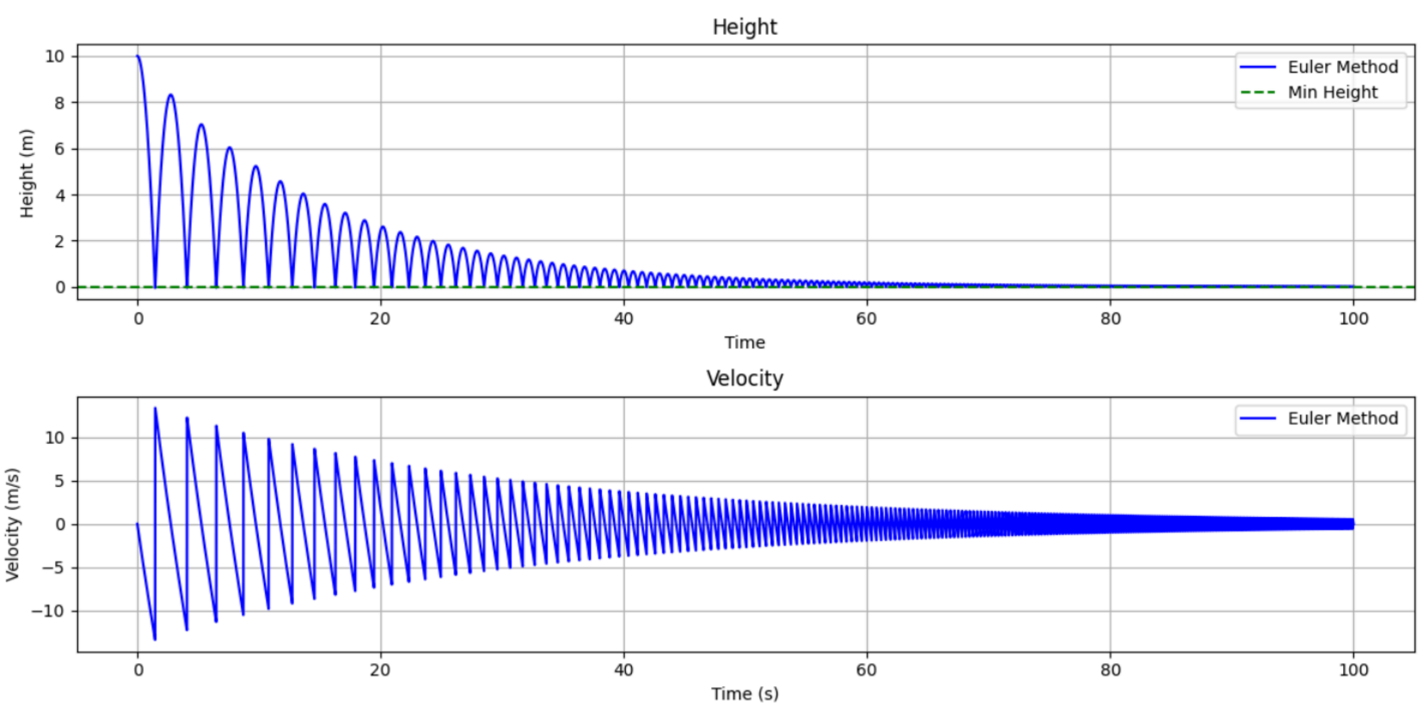
Se obtuvieron resultados que exhiben a la pelota rebotando por una mayor cantidad de tiempo. Esta tarda una mayor cantidad de unidades de tiempo desde ser tirada hasta llegar a su estado de reposo.

3. **Tercer experimento:** variación del coeficiente de rebote

En primer lugar se varió el coeficiente de rebote, este fue aumentado a $b = 50$, con los demás parámetros por defecto, y se obtuvo el siguiente resultado:



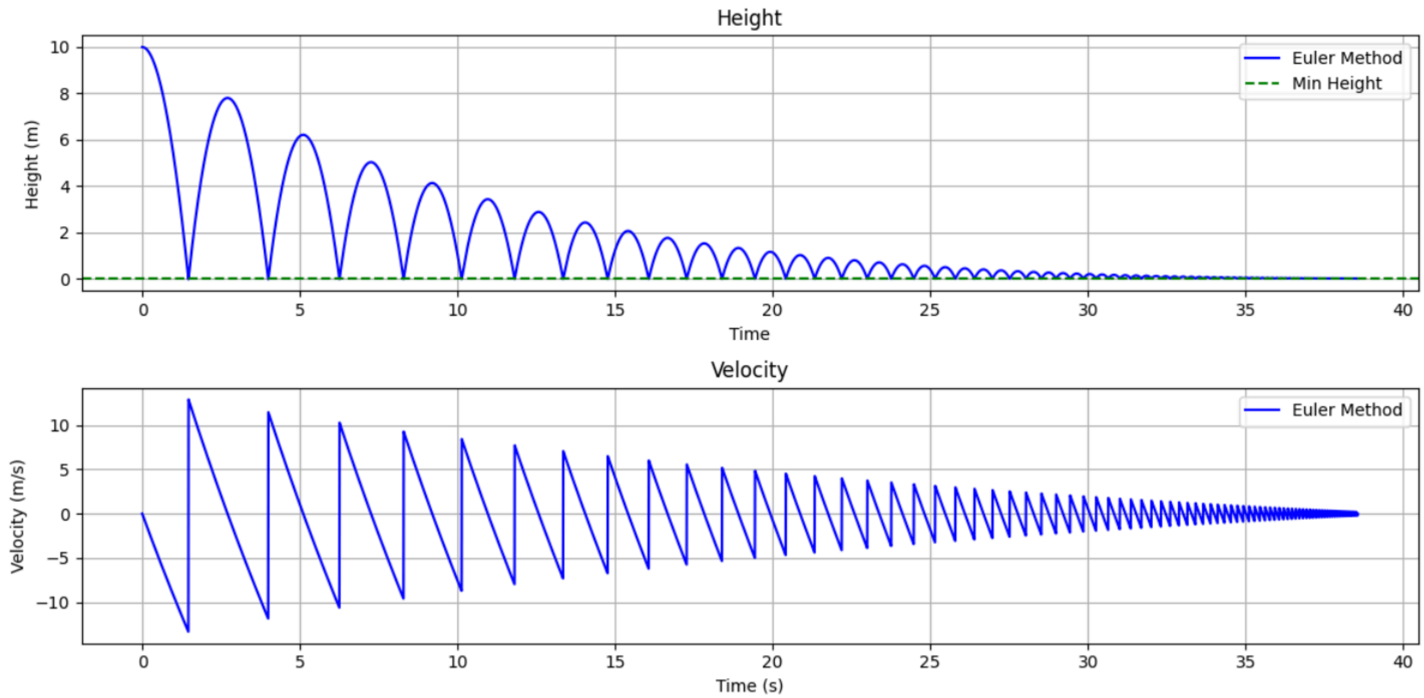
Se puede observar que la pelota rebota una menor cantidad de veces y a su vez alcanza una menor velocidad. Esto produce que la pelota se detenga por completo antes que en el experimento inicial.
Luego se disminuyó este parámetro, tomando a $b = 10$.



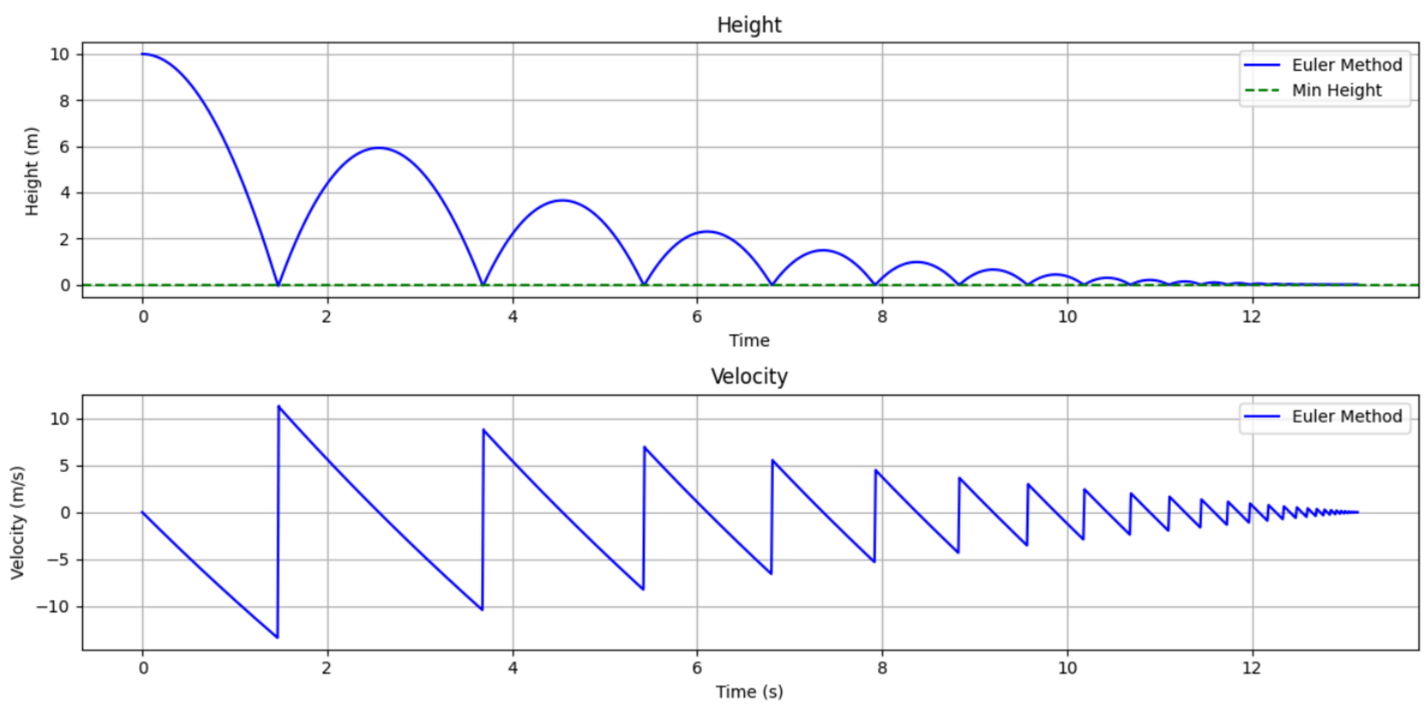
Se puede observar que la pelota realiza una mayor cantidad de rebotes por un intervalo de tiempo mayor.

4. **Cuarto experimento:** variación del parámetro k.

Se tomó $k = 200000$ la cual es la constante elástica del suelo, representa su "dureza".



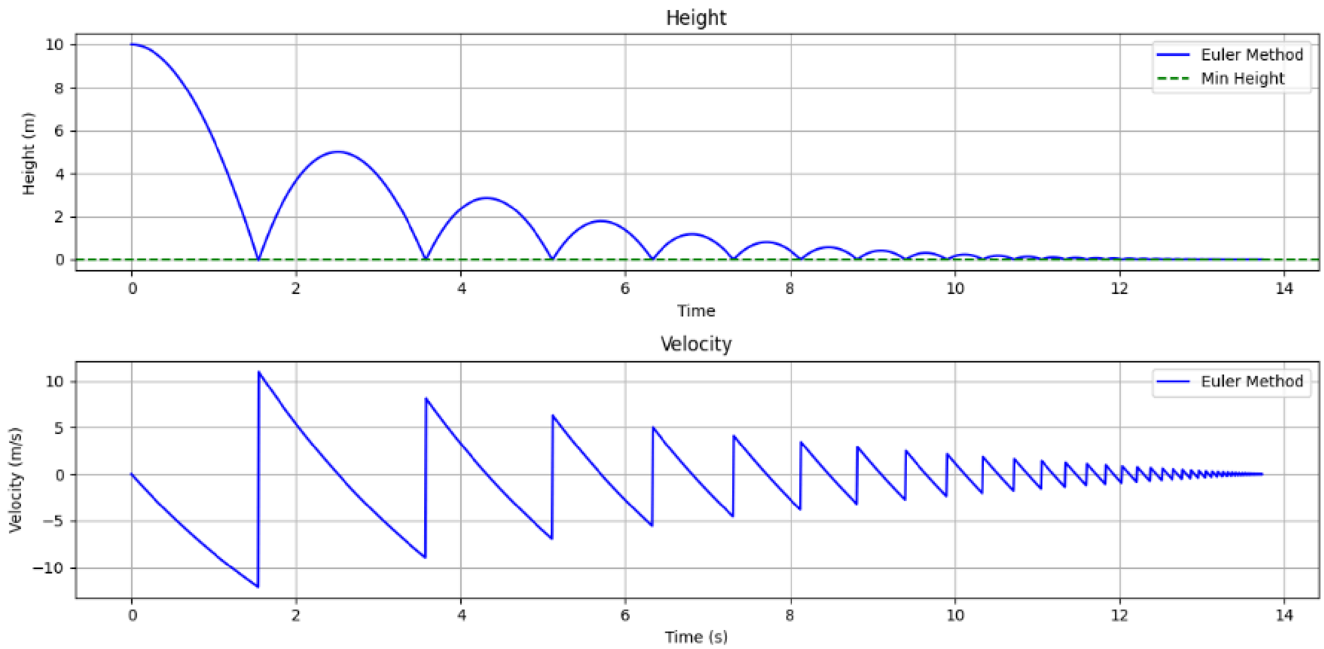
Con un k mayor al dado inicialmente la pelota rebota una mayor cantidad de veces alcanzando una mayor velocidad.
Luego, se disminuyó esta constante tomando a $k = 50000$.



Al ser más “blando” el suelo, la pelota rebota menos y por una menor cantidad de tiempo.
Se puede concluir en base a este experimento que a mayor dureza del suelo, mayor es la capacidad de rebote de la pelota en este modelo.

5. **Quinto experimento:** variación del parámetro b_a.

Se tomó el siguiente parámetro: $b_a = 0.3$, el cual es el coeficiente de rozamiento aéreo.



Se notó que con un mayor coeficiente de rozamiento aéreo la pelota pierde velocidad más rápidamente, provocando que rebote una menor cantidad de veces.

Modelado del problema: (método trapezoidal)

A su vez, se llevó a cabo la implementación de un método de segunda clase, el método Trapezoidal, con el objetivo de poder comparar sus resultados con los obtenidos mediante el método Euler.

- Cálculo de la altura:

$$x(t + h) = x(t) + 0.5 * h\left[\frac{dx(t+h)}{dt} + \frac{dx(t)}{dt}\right]$$

$$x(t + h) = x(t) + 0.5 * h\left[v(t + h) + v(t)\right]$$

- Cálculo de la velocidad:

$$v(t + h) = v(t) + 0.5 * h\left[\frac{dv(t+h)}{dt} + \frac{dv(t)}{dt}\right]$$

$$v(t + h) = v(t) + 0.5 * h\left[a(t + h) + a(t)\right]$$

La implementación de dicho método es la siguiente:

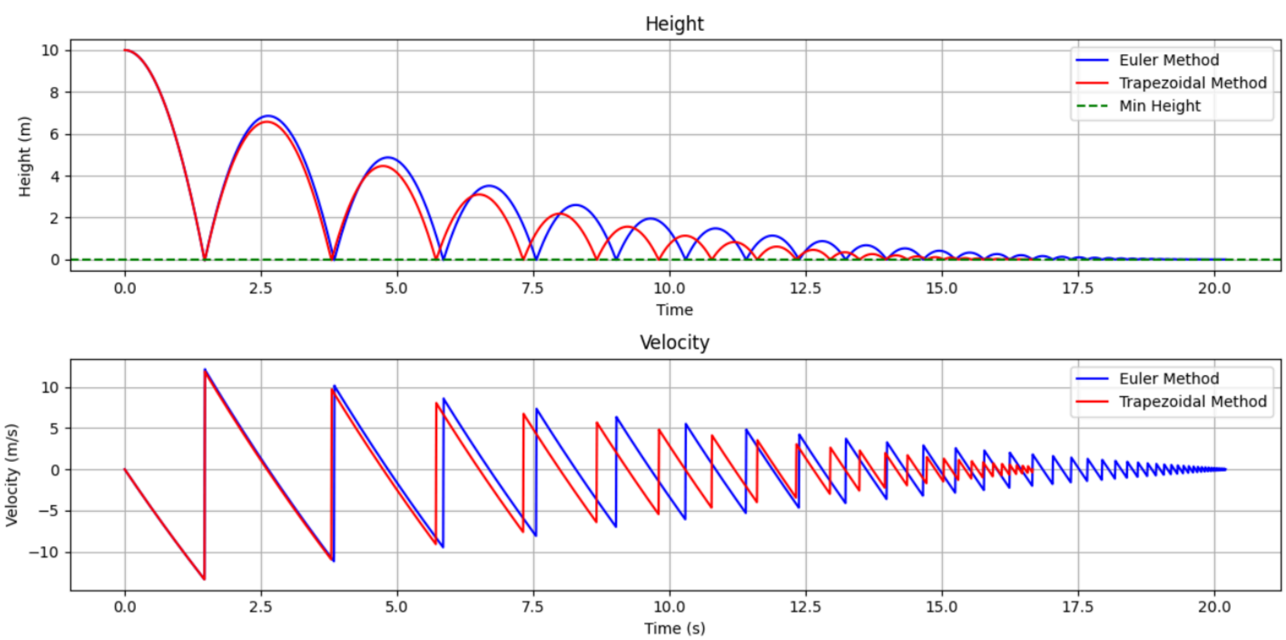
```
from integration_methods.euler import EulerMethod

class TrapezoidalMethod:

    def __init__(self, h):
        self.h = h
        self.euler_method = EulerMethod(h)
```

```
    def compute_next_height(self, prev_height, prev_velocity, ba, m , b, g, k):
        result = prev_height + 0.5 * self.h * (prev_velocity + self.euler_method.compute_next_velocity(prev_height, prev_velocity, ba, m , b, g, k))
        return result

    def compute_next_velocity(self, prev_height, prev_velocity, ba, m , b, g, k):
        result = prev_velocity + 0.5 * self.h * (self.get_acceleration(prev_height, prev_velocity, ba, m , b, g, k) + self.euler_method.get_acceleration(prev_height, prev_velocity, ba, m , b, g, k))
        return result

    def get_acceleration(self, prev_height, prev_velocity, ba, m , b, g, k):
        if prev_height > 0:
            return -ba / m * prev_velocity - g
        else:
            return -k / m * prev_height - b / m * prev_velocity - g
```

Una vez implementados estos dos métodos, se ejecutó la simulación utilizando ambos para poder comparar sus resultados. Con los mismos parámetros para ambas simulaciones:

$v(0) = 0,\ x(0) = 10,\ ba = 0.1,\ m = 1,\ b = 30,\ g = 9.8$, $k = 10000$ y $h = 0.0001$

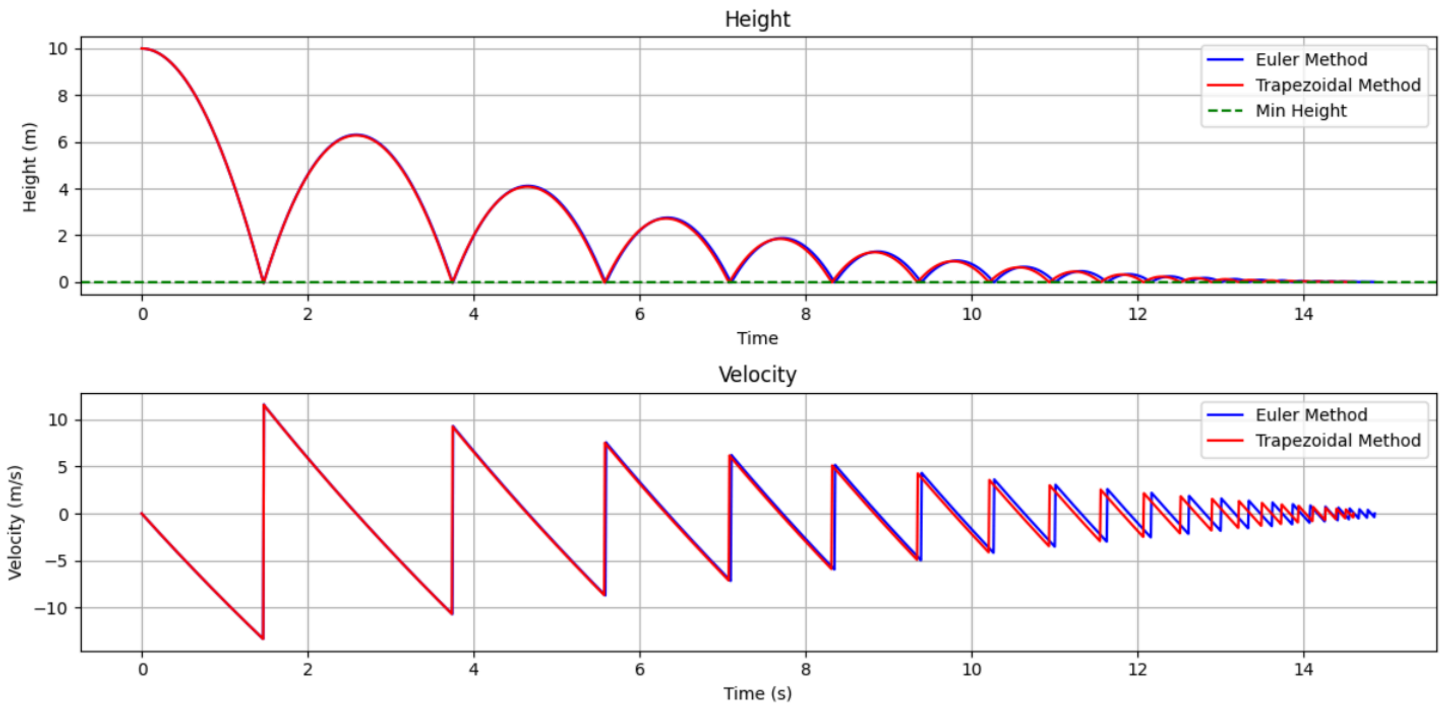
Se obtuvieron los siguientes resultados:



Se puede notar una leve diferencia entre ambos, daría la impresión de que las simulaciones comienzan igual o con diferencias imperceptibles en el gráfico a simple vista, pero luego de la primera vez que la pelota toca el suelo, el comportamiento de estas difiere.

No se cuenta con una solución analítica con la que comparar para poder determinar qué método fue el más aproximado, pero este gráfico permite exhibir la diferencia presente en los resultados obtenidos por ambos métodos.

Sin embargo, si se modifica el paso de integración h, disminuyendolo, se puede observar cómo los gráficos de ambos métodos se aproximan entre sí. Con $h = 0.00001$ se obtienen los siguientes resultados:



La diferencia entre ambos gráficos es mucho menor, logrando una mayor precisión en los resultados de la simulación.