



Caída de la servilleta, análisis por el método de Euler y el método de Runge-Kutta

Carlos Mario Álvarez Lizcano - Física

Facultad de Ciencias, Escuela de Física

Universidad Industrial de Santander

Índice

1. Resumen	1
2. introducción	1
3. Marco teórico	2
3.1. Segunda ley de Newton	2
3.2. Fricción despreciable	2
3.3. Fricción considerable	2
3.4. Método de Euler	3
3.5. Método de Runge-Kutta	3
4. Metodología	4
5. Resultados	5
5.1. Resultado por Newton	5
5.2. Resultado por Euler	6
5.3. Resultado por Runge-Kutta	7
6. Conclusiones y recomendaciones	7

1. Resumen

En este informe, se analizará el método de la mecánica clásica para analizar la caída de una servilleta, y se comparará dicho método con métodos más modernos, dos en particular, el método de Euler y el método de Runge Kutta, para después analizar cuál de los diferentes métodos es el más eficiente y el que mejor se ajusta a la situación.

2. introducción

El análisis de fuerzas y gravedad es fundamental para comprender el movimiento de los objetos en el mundo físico. Desde la formulación de la segunda ley de Newton hasta los métodos numéricos modernos, los científicos han desarrollado diferentes enfoques para abordar este tema. En este informe, nos centraremos en analizar un método clásico de análisis de fuerzas y gravedad, basado en la segunda ley de Newton, en dos escenarios específicos: cuando la fricción del aire es despreciable y cuando es significativa, y comparar este método con el método de Euler y el método de Runge-Kutta, para determinar cuál es más fácil, óptimo y preciso en la situación de la caída de la servilleta.

3. Marco teórico

3.1. Segunda ley de Newton

La segunda ley de Newton, también conocida como la ley de la fuerza y la aceleración, es uno de los principios fundamentales de la física clásica. Establece la relación entre la fuerza aplicada a un objeto y la aceleración que experimenta. Según la segunda ley de Newton, la aceleración de un objeto es directamente proporcional a la fuerza neta que actúa sobre él e inversamente proporcional a su masa.

La segunda ley de Newton es de suma importancia en el estudio de la dinámica, ya que nos permite comprender cómo las fuerzas actúan sobre los objetos y cómo se relacionan con sus cambios en movimiento. Esta ley es aplicable a una amplia gama de situaciones, desde el movimiento de los cuerpos en caída libre hasta el movimiento de los planetas en el espacio.

La segunda ley de Newton, y la ecuación que se utiliza en este trabajo es la siguiente:

$$ma = \sum_i F_{ext} \Rightarrow ma = mg - \kappa v, \quad (1)$$

Donde m es la masa del cuerpo que cae, a es la aceleración del cuerpo, g es la gravedad, en este caso, del planeta tierra, y k será una constante de rozamiento la cuál frenará al objeto que cae, cabe aclarar que la constante de rozamiento o fricción no es constante, pues a medida que aumenta la velocidad de caída gracias a la aceleración, la constante k también aumenta, pues esta se opone al movimiento.

Para la caída libre, hay dos casos a tomar en cuenta:

3.2. Fricción despreciable

Cuando la fricción es muy diminuta a tal punto que su consideración no genera un cambio significativo en la simulación, entonces podemos decir que es despreciable, pues es debido a que el objeto que cae tiene muy poca área superficial para que la fricción sea notable o tiene demasiado peso.

Modificando la fórmula en base a esta hipótesis, tendríamos lo siguiente:

$$ma = \sum_i F_{ext} \Rightarrow ma = mg \Rightarrow a = g. \quad (2)$$

y además, como es una caída libre sencilla, tendríamos también las siguientes fórmulas:

$$v_f = v_0 + gt \quad y \quad d = v_0 t + g \frac{t^2}{2}, \quad (3)$$

3.3. Fricción considerable

En el caso alternativo en el que la fuerza de rozamiento, en este caso, del aire, sea notable, entonces la tendremos que tomar en cuenta, y por lo tanto, la caída ya no será constante, pues la aceleración no será constante.

Pero como por el momento se debe ser recursivo y usar las fórmulas (3), entonces lo que se puede hacer es dividir la caída en intervalos de tiempos para que estas fórmulas puedan ser aplicables, pues la variación de la aceleración en un intervalo de tiempo pequeño es despreciable, esto se refiere a:

$$[t_0, t_f] = [t_0, t_1] \cup [t_1, t_2] \cup [t_2, t_3] \cup \dots \cup [t_i, t_{i+1}] \cup \dots \cup [t_{N-2}, t_{N-1}] \cup [t_{N-1}, t_N = t_f],$$

Por lo que para cada intervalo pequeño de tiempo, tenemos la siguiente velocidad y distancia:

$$[t_i, t_{i+1}]: \left. \begin{array}{l} v(t_i) = v_i \\ x(t_i) = x_i \end{array} \right\} \Rightarrow \left. \begin{array}{l} v_{i+1} = v_i + a(x_i, v_i, t_i)[t_{i+1} - t_i] \\ x_{i+1} = x_i + v_i[t_{i+1} - t_i] + a(x_i, v_i, t_i) \frac{[t_{i+1} - t_i]^2}{2}; \end{array} \right.$$

hasta $i = N - 1$:

$$[t_{N-1}, t_N]: \left. \begin{array}{l} v(t_{N-1}) = v_{N-1} \\ x(t_{N-1}) = x_{N-1} \end{array} \right\} \Rightarrow \left. \begin{array}{l} v_N = v_{N-1} + a(x_{N-1}, v_{N-1}, t_{N-1})[t_N - t_{N-1}] \\ x_N = x_{N-1} + v_{N-1}[t_N - t_{N-1}] + a(x_{N-1}, v_{N-1}, t_{N-1}) \frac{[t_N - t_{N-1}]^2}{2} \end{array} \right.$$

Estos intervalos vendrán de mil maravillas para analizar los siguientes métodos.

3.4. Método de Euler

El método de Euler es una técnica utilizada para aproximar soluciones de ecuaciones diferenciales ordinarias (EDOs) de primer orden. Se basa en la idea de discretizar la variable independiente y aproximar los cambios en la variable dependiente a través de una serie de pasos pequeños.

La idea central detrás del método de Euler es descomponer una EDO en un conjunto de ecuaciones diferenciales más pequeñas y manejables, que describen cómo cambia la variable dependiente a medida que se avanza en el tiempo. A partir de un valor inicial conocido, el método de Euler utiliza una aproximación lineal para estimar la pendiente en ese punto y, a partir de ahí, calcular el valor aproximado de la variable dependiente en el siguiente paso de tiempo. Esto más adelante vendrá útil pues se descomponerá la caída libre en intervalos de tiempo para que se puedan usar las fórmulas lineales.

El método de Euler consiste en lo siguiente:

Dividir el intervalo que empieza en x_0 y termina en x_f en intervalos n de ancho h :

$$h = \frac{x_f - x_0}{n}$$

entonces en el intervalo $[x_0, x_f]$ tenemos x_{n+1} elementos, $x_0, x_1, x_2, \dots, x_n$, para los que se cumple lo siguiente:

$$x_i = x_0 + ih \quad 0 \leq i \leq n$$

Entonces si tenemos una condición inicial $y(x_0) = y_0$ esto representa un punto $P_0 = (x_0, y_0)$ por la cual pasa la curva solución del problema inicial, la cuál se llamará $F(x) = y$, teniendo este punto P_0 podemos derivar F en ese punto, por lo que:

$$F'(x) = \frac{dy}{dx}(P_0) = f(x_0, y_0)$$

Y esto es una pendiente, por lo que se puede trazar una recta que pasa por el punto (x_0, y_0) con pendiente $f(x_0, y_0)$, y esta recta reemplaza a $F(x)$, por lo que en ella se ubica el valor de y correspondiente a x_1 . por lo que podemos tener la siguiente fórmula:

$$\frac{y_1 - y_0}{x_1 - x_0} = f(x_0, y_0)$$

por lo que si despejamos a y_1 tenemos que:

$$y_1 = y_0 + (x_1 - x_0)f(x_0, y_0) \Rightarrow y_1 = y_0 + hf(x_0, y_0)$$

por lo que generalizando esta fórmula, tenemos que:

$$y_n = y_{n-1} + hf(x_{n-1}, y_{n-1}) \quad (4)$$

Por lo que así estimaremos una solución a la curva que desconocemos, cabe destacar que el error disminuye mientras más intervalos se tomen, además de que este método es de primer orden, por lo que el error local depende del cuadrado del tamaño del intervalo, y el error global es proporcional al tamaño del intervalo.

3.5. Método de Runge-Kutta

El método, o mejor dicho, métodos de Runge-Kutta, como el método de Euler, son familias de métodos los cuales son cíclicos y que pueden ser tanto implícitos como explícitos, para aproximar las soluciones de EDOs. tiene múltiples pasos que utilizan una combinación de pendientes para estimar los cambios en la variable dependiente

a medida que se avanza en el tiempo. La diferencia entre este método y el método de Euler radica en que este último utiliza solo una pendiente en cada paso, pero el método de Runge-Kutta mejora la precisión de la aproximación al considerar múltiples pendientes a lo largo de un intervalo de tiempo.

En su forma general, el método de Runge-Kutta tiene la siguiente forma:

$$y_{n+1} = y_n + h \sum_{i=1}^s b_i k_i \quad (5)$$

Donde h es el paso por ciclo, o en otras palabras el incremento de Δt_n entre los puntos t_n y t_{n+1} , y_n es el valor conocido de la solución en el punto actual de evaluación. En la primera iteración, este valor suele ser conocido o inicializado, k_i representa las pendientes de la función en diferentes puntos dentro del intervalo de integración, y_n es el valor conocido de la función en el punto actual y y_{n+1} es el valor aproximado de la función en el siguiente punto. Las pendientes en cada punto tienen su forma de calcularse, y es la siguiente:

$$\begin{aligned} k_1 &= f(t_n, y_n) \\ k_2 &= f(t_n + c_2 h, y_n + h(a_{21} k_1)) \\ k_3 &= f(t_n + c_3 h, y_n + h(a_{31} k_1 + a_{32} k_2)) \\ k_4 &= f(t_n + c_4 h, y_n + h(a_{41} k_1 + a_{42} k_2 + a_{43} k_3)) \\ &\vdots \\ k_s &= f(t_n + c_s h, y_n + h(a_{s1} k_1 + a_{s2} k_2 + \dots + a_{s(s-1)} k_{s-1})) \end{aligned}$$

Cabe aclarar que los coeficientes a_{ij} y b_i son parámetros específicos del método de Runge-Kutta que determinan cómo se combinan las pendientes para obtener una mejor aproximación. Los valores de estos coeficientes varían dependiendo del orden del método de Runge-Kutta que se esté utilizando.

4. Metodología

Como ya está clara la teoría básica de cada método (Newton, Euler, Runge-Kutta), se empezará a implementar la implementación práctica. para ello y como este informe es netamente teórico, se desarrollará un código en lenguaje de Python que permita simular el movimiento de un objeto bajo la influencia de la gravedad y las fuerzas en los dos escenarios de fricción del aire: despreciable y significativa.

El usuario podrá ingresar los valores que quiera, como por ejemplo la masa, si se quiere, de la servilleta, altura a la que cae, y también se puede modificar el código para cambiar el medio, como por ejemplo alterar la gravedad o la resistencia del aire.

La comparación entre los métodos se realizará mediante el análisis de los resultados obtenidos. Se evaluará la precisión de cada método mediante la comparación de las gráficas calculadas con los valores teóricos. Asimismo, se analizará la eficiencia computacional de cada método, considerando el tiempo de ejecución y los recursos necesarios para obtener los resultados.

Finalmente, se presentarán y discutirán los hallazgos obtenidos en la comparación de los métodos. Se destacarán las fortalezas y debilidades de cada enfoque en relación con la precisión y eficiencia. Además, se proporcionarán conclusiones sobre cuándo es apropiado utilizar cada método según el escenario de fricción del aire y las características del sistema estudiado.

Los códigos se encontrarán disponibles como archivo .py en el repositorio de tareas en GitLab, así el que quiera los podrá correr con sus propias mediciones.

5. Resultados

A continuación, se mostrará los resultados arrojados por cada código, cada uno con su método específico:

5.1. Resultado por Newton

El tiempo de ejecución promedio de este código es de 4943300 nanosegundos, el código se corrió con una altura de 2 metros, una servilleta que pesa 5 gramos y cada lado tiene 5 centímetros de longitud, proporcionando la siguiente gráfica:

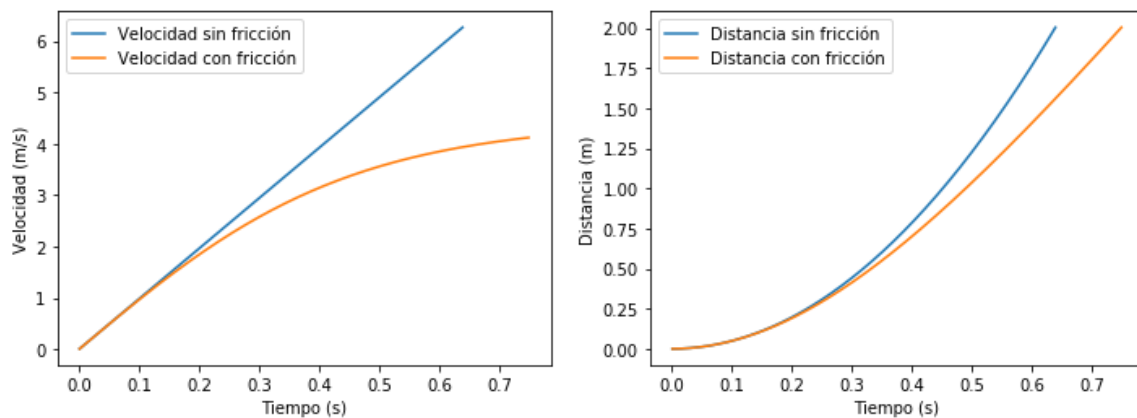


Figura 1: Comparación de la velocidad y la distancia recorrida con fricción y sin fricción

Los datos tienden a esa organización (ya sean más suaves o burdas las curvas), independientemente del dato que se le introduzca al código, pues como se puede ver, la velocidad con fricción tiende a ser una línea horizontal, puesto que se va frenando hasta ser constante, igualando fuerzas con la resistencia del aire, y la distancia recorrida con fricción tiende a una línea recta, puesto que, a medida que pasa el tiempo, la velocidad empieza a ser constante, y por lo tanto la distancia recorrida deja de crecer de manera exponencial y empezará a crecer de manera lineal.

5.2. Resultado por Euler

El tiempo de ejecución promedio usando este método es de 167241900 nanosegundos, siendo más lento que el anterior, además, arroja la siguiente gráfica, con resistencia de 0.1 y la masa de 5 gramos (los mismos valores para el código anterior), cabe recalcar que no se tomó en consideración analizar la caída sin fricción, pues daría como resultado valores lineales esperados y predecibles: Analizando las gráficas, podemos ver que son prácticamente lo

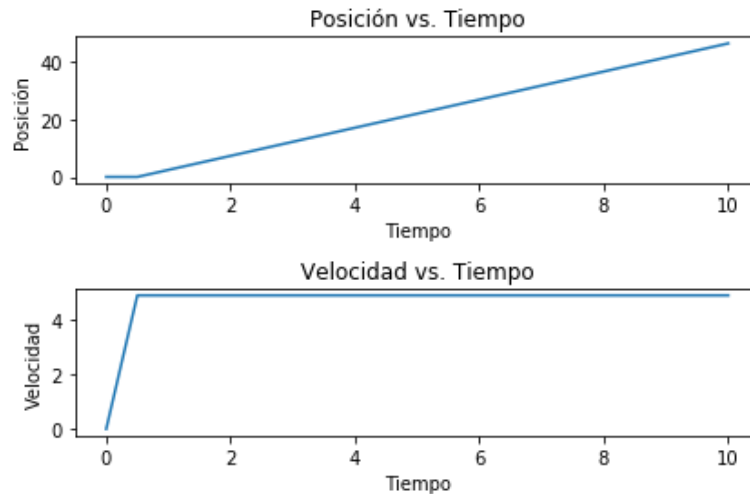


Figura 2: Comparación de la posición y el tiempo, y la velocidad y el tiempo

mismo que las gráficas arrojadas por el método de Newton, solo que mucho más burdas y menos sofisticadas, esto es consecuencia del método de Euler, pues es poco preciso y por decirlo de alguna manera "corta esquinas". Pero este método da los suficientes resultados para hacer el mismo análisis; la velocidad va creciendo hasta que llega a la igualdad con la resistencia del medio, y a partir de ahí se vuelve constante, y eso se ve reflejado con la distancia recorrida, aunque no se vea muy bien, pues como ya se mencionó, el método de Euler es poco práctico, la distancia recorrida al principio va creciendo exponencialmente, como una curva, pero al final se vuelve lineal, yendo a la par con la linealidad de la velocidad.

5.3. Resultado por Runge-Kutta

El tiempo de ejecución por este método es de 174534100 nanosegundos, siendo el más lento de los tres métodos, pero el lado positivo es la gráfica que arroja este método, la cuál es la siguiente: Como podemos ver, la gráfica es

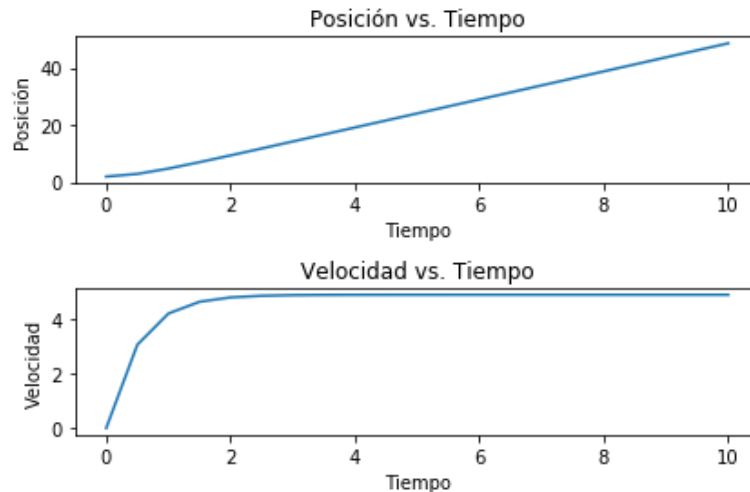


Figura 3: Comparación de la posición y el tiempo por medio de Runge-Kutta, y la velocidad y el tiempo, también por Runge-Kutta

casi que idéntica a la que el método de Euler arrojó, pero esta es más refinada, con mucho más detalle, haciendo el análisis de la velocidad y la distancia mucho más fácil.

Ahora podemos ver que la velocidad se va frenando debido a la resistencia del aire hasta llegar al equilibrio, y lo mismo ocurre con la distancia recorrida, se puede apreciar una pequeña curva al inicio debido a la velocidad creciendo, pero después se vuelve lineal.

6. Conclusiones y recomendaciones

Después de haber realizado comparaciones, experimentación, construcción y refinación de código, se puede concluir que, aunque los tres métodos son viables, funcionan, y describen la cinemática de la caída libre, el mejor método para analizar el problema de la servilleta es el método de Newton.

Esto es gracias a su simplicidad, la construcción del código del método de Newton fue relativamente sencilla, además de que este método no consume tantos recursos computacionales a la hora de hacer pruebas, y por último, proporciona resultados precisos y fáciles de entender.

El método de Euler es el menos eficaz para el problema de la servilleta, sus resultados, como se mencionó antes, son muy imprecisos en el aspecto de que se salta bastantes detalles, y también consume más recursos computacionales para dar resultados que no son tan buenos.

El método de Runge-Kutta es bueno, pero es el que consume más recursos, no obstante sus resultados son detallados y mejores, es como una versión más pulida del método de Euler, pero aún así considero que el método de Runge-kutta para el problema de la servilleta es muy desproporcionado, como matar a una mosca con una escopeta, en fin, este método es bastante bueno pero debería de usarse para situaciones más complejas.