

PROYECTO FINAL DE ECUACIONES DIFERENCIALES ORDINARIAS

ODES
INTEGRANTES:
ORDINARY
DIFFERENTIAL
JUAN CARLOS CARMENATE DÍAZ

ADIÁN MIGUEL SANAMÉ LEÓN

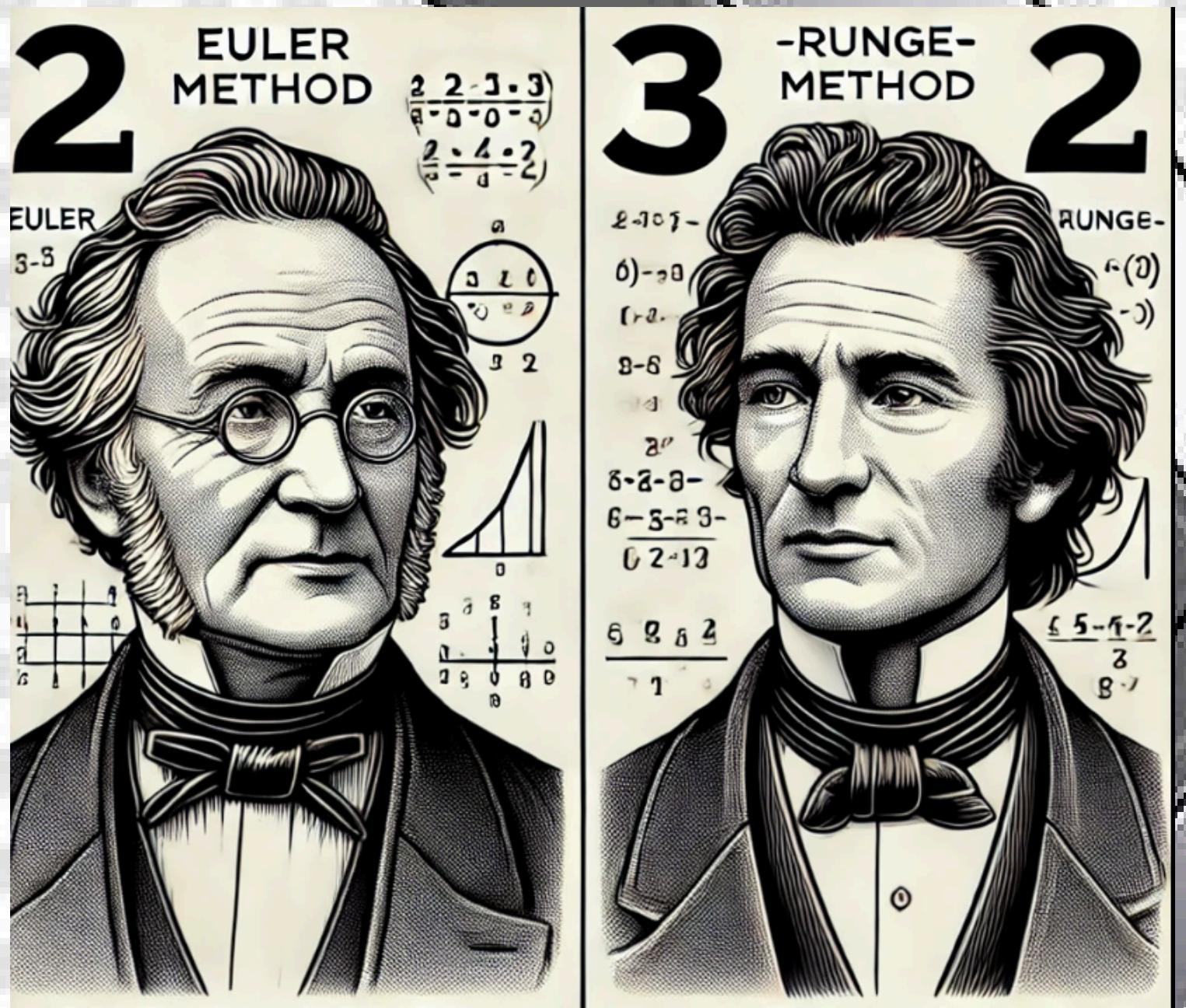
SEBASTIAN GONZÁLES ALFONSO

JORGE LUIS HERRERA CECILIA

Euler

Vs

RK



Who is better ?

Comparación

Este proyecto tiene como objetivo realizar una comparación entre el método de Euler mejorado y el método de Runge-Kutta más utilizado por su versatilidad, el de 4to orden.

f(x,y)

x0

y0

Paso h

Cantidad de pasos

Resolver

Graficar

Mostrar Precisión

Mostrar Tabla

Mostrar Interpolación de Newton

Objetivo Final :

Para lograr una comparación exitosa usaremos una aplicación graficadora de las funciones soluciones de EDO`s introducidas ,que muestren ambos métodos junto a la solución real explícita en caso de que pueda hallarse analíticamente .

f(x,y) x0 y0 Paso h Cantidad de pasos

Resolver

Graficar

Mostrar Precisión

Mostrar Tabla

Mostrar Interpolación de Newton

Funcionamiento

Cada botón es una gráfica o tabla que realiza la operación señalada

comparación entre E-M y RK4

isoclinas

Precisión

Tabla de errores

Gráfica de interpolación

De manera general
Nuestro equipo diseñó a
través de cada 1 de los
métodos una serie de
pasos para la resolución
de estos de manera
rápida y eficiente

1_Diseño teórico

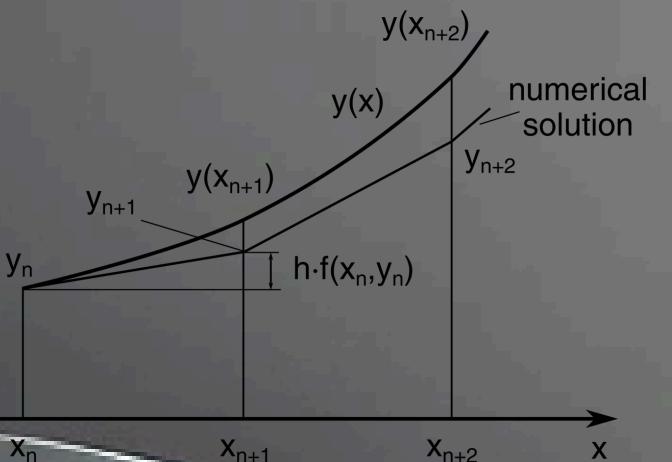
2_Fórmula

3_Código óptimo

4_Manejo de gráficas

5_Cálculo de errores

Algunos pasos del diseño preeliminar



Autoguardado - Euler Mejorado - Guardado en Este PC - Buscar ISABELLA CAROLINA PINEROS ARIZA

Solución de Euler Mejorado:

$$f(x, y) = \frac{dy}{dx}$$
$$y_{i+1}^* = y_i + f(x_i, y_i)h$$
$$y_{i+1} = y_i + h \left[\frac{f(x_i, y_i) + f(x_{i+1}, y_{i+1}^*)}{2} \right]$$
$$x_{i+1} = x_i + h$$

Página 1 de 1 47 páginas Espanol (Colombia)

1_Estudio teórico.

2_Correctitud y plantamiento de fórmula.

3_Transformación a código

Tabnine | Edit | Test | Explain | Document

```
def euler_improved(f, x0, y0, h, n):
    x_values, y_values = [x0], [y0]

    for i in range(n):
        x = x_values[-1]
        y = y_values[-1]

        k1 = f(x, y) # Calcula la pendiente en el punto inicial
        y_predict = y + h * k1 # Predice el valor de y en el siguiente paso

        k2 = f(x + h, y_predict) # Calcula la pendiente en el punto predicho
        y_corrected = y + (h / 2) * (k1 + k2) # Corrige el valor de y usando el promedio de las pendientes

        x_values.append(round(x + h, 3)) # Agrega el nuevo valor de x a la lista
        y_values.append(y_corrected) # Agrega el nuevo valor de y a la lista

    return x_values, y_values
```

Errores numéricos:

En los métodos numéricos para resolver EDOs, el error surge debido a la aproximación de funciones continuas en puntos discretos. Se identifican tres tipos principales de error:

1-Error de Discretización: Es la diferencia entre la solución exacta de la EDO y la aproximación numérica. Se genera porque los métodos numéricos, como Runge-Kutta, solo evalúan la función en puntos discretos en lugar de seguir la solución exacta.

2-Error Local de Truncamiento (ELT): Es el error en un solo paso del método, suponiendo que la solución exacta es conocida al inicio del paso. Se calcula como la diferencia entre la solución exacta en el siguiente punto y la aproximación obtenida con el método numérico.

3-Error Global: Representa la acumulación de errores locales en todos los pasos hasta llegar al punto final. Depende tanto del número de pasos como del tamaño del paso utilizado.

Profundizando errores:

El método de Runge-Kutta tiene un orden de convergencia p , determinado al comparar su expansión en serie de Taylor con la de la solución exacta.

Error Local de Truncamiento (ELT): Es proporcional a $O(h^{(p+1)})$ y representa el error en un solo paso del método.

Error Global: Se acumula a lo largo de los pasos y es proporcional a $O(h^p)$, ya que el número de pasos es inversamente proporcional al tamaño del paso h .

Fórmulas

El error local de truncamiento para un método de Runge-Kutta de orden p es:

$$\tau_{n+1} = Ch^{p+1} + \mathcal{O}(h^{p+2}),$$

donde C es una constante que depende de la EDO y del método.

El error global, después de N pasos con tamaño de paso h , es:

$$E_N = C'h^p + \mathcal{O}(h^{p+1}),$$

donde C' es otra constante que depende de la EDO y del método.

Esta expresión representa la acumulación de los errores locales a lo largo de todos los pasos del método. Aunque el error local es $\mathcal{O}(h^{p+1})$, el error global es $\mathcal{O}(h^p)$.

Una aproximación más profunda

Supongamos que integramos la EDO desde x_0 hasta $x_N = x_0 + Nh$, donde N es el número de pasos. En cada paso, se comete un error local:

$$\tau_{n+1} = \mathcal{O}(h^{p+1}).$$

Después de N pasos, el error global E_N es la suma de los errores locales:

$$E_N = \sum_{n=0}^{N-1} \tau_{n+1}.$$

El número de pasos N está relacionado con el tamaño del paso h y el intervalo de integración $T = x_N - x_0$:

$$N = \frac{T}{h}.$$

Por lo tanto, el error global es:

$$E_N = N \cdot \mathcal{O}(h^{p+1}) = \frac{T}{h} \cdot \mathcal{O}(h^{p+1}) = \mathcal{O}(h^p).$$

Aunque el error local es $\mathcal{O}(h^{p+1})$, el error global es $\mathcal{O}(h^p)$ porque el número de pasos N es inversamente proporcional a h . Esto significa que, al reducir h , el error global disminuye más lentamente que el error local. Sin embargo, como la constante C es desconocida, se hace necesario buscar una forma computable de calcular el error. Una forma muy utilizada en la práctica es el doble cómputo, que consiste en estimar la función en un punto, primero con un paso de tamaño h , y luego de tamaño $h/2$ (de ahí doble cómputo). Se procede de la siguiente forma:

1. Se expresa el error en función de h :

$$\tau_{i+1} = y(x_{i+1}) - y_h(x_{i+1}) = C(x_{i+1})h^{p+1} + \mathcal{O}(h^{p+2}).$$

2. Se expresa el error en función de $h/2$:

$$\tau_{i+1} = y(x_{i+1}) - y_{h/2}(x_{i+1}) = C(x_{i+1})\left(\frac{h}{2}\right)^p + \mathcal{O}\left(\left(\frac{h}{2}\right)^{p+1}\right).$$

3. Se restan las expresiones y se despeja la constante:

$$C(x_{i+1}) \approx \frac{2^p (y_{h/2}(x_{i+1}) - y_h(x_{i+1}))}{(2^p - 1)h^p}.$$

4. Se sustituye el valor de C en la expresión inicial:

$$\tau_{i+1} \approx \frac{2^p (y_{h/2}(x_{i+1}) - y_h(x_{i+1}))}{2^p - 1}.$$

Esta última expresión constituye una estimación del error global de la solución aproximada y_h en el punto x , es computable y no requiere la evaluación de C .

Comparación E-M y RK4,semejanzas

- **Incremento Aproximado:**

Ambos métodos calculan la nueva ordenada y_{n+1} sumando a la ordenada anterior y_n un incremento Δy_n . Este incremento se formula para que, al expandirse en serie de Taylor, coincida con la expansión de la solución exacta $y(x_n + h)$ hasta un determinado orden, utilizando únicamente la primera derivada f (sin requerir derivadas de orden superior).

- **Construcción del Incremento:**

En lugar de usar directamente las derivadas (como en el método de Taylor), los métodos de Runge-Kutta construyen el incremento Δy_n como una combinación lineal de evaluaciones de f en varios puntos dentro del intervalo $[x_n, x_n + h]$. Esta formulación se expresa en forma general:

$$y_{n+1} = y_n + h \sum_{i=1}^s b_i K_i,$$

donde cada K_i se define como:

$$K_i = f\left(x_n + c_i h, y_n + h \sum_{j=1}^s a_{ij} K_j\right), \quad i = 1, 2, \dots, s.$$

Los coeficientes a_{ij} , b_i y c_i (generalmente organizados en el arreglo de Butcher) determinan completamente el método.

Particularidades

1. Método de Euler ($s = 1, b_1 = 1$):

$$y_{n+1} = y_n + h f(x_n, y_n)$$

Fórmula de orden 1 basada en la regla de los rectángulos

$$\int_a^b f(x) dx \approx \sum_{i=0}^{n-1} f(x_i) \Delta x.$$

2. Método de Euler mejorado ($s = 2, b_1 = 0, b_2 = 1, a_{21} = c_2 = \frac{1}{2}$):

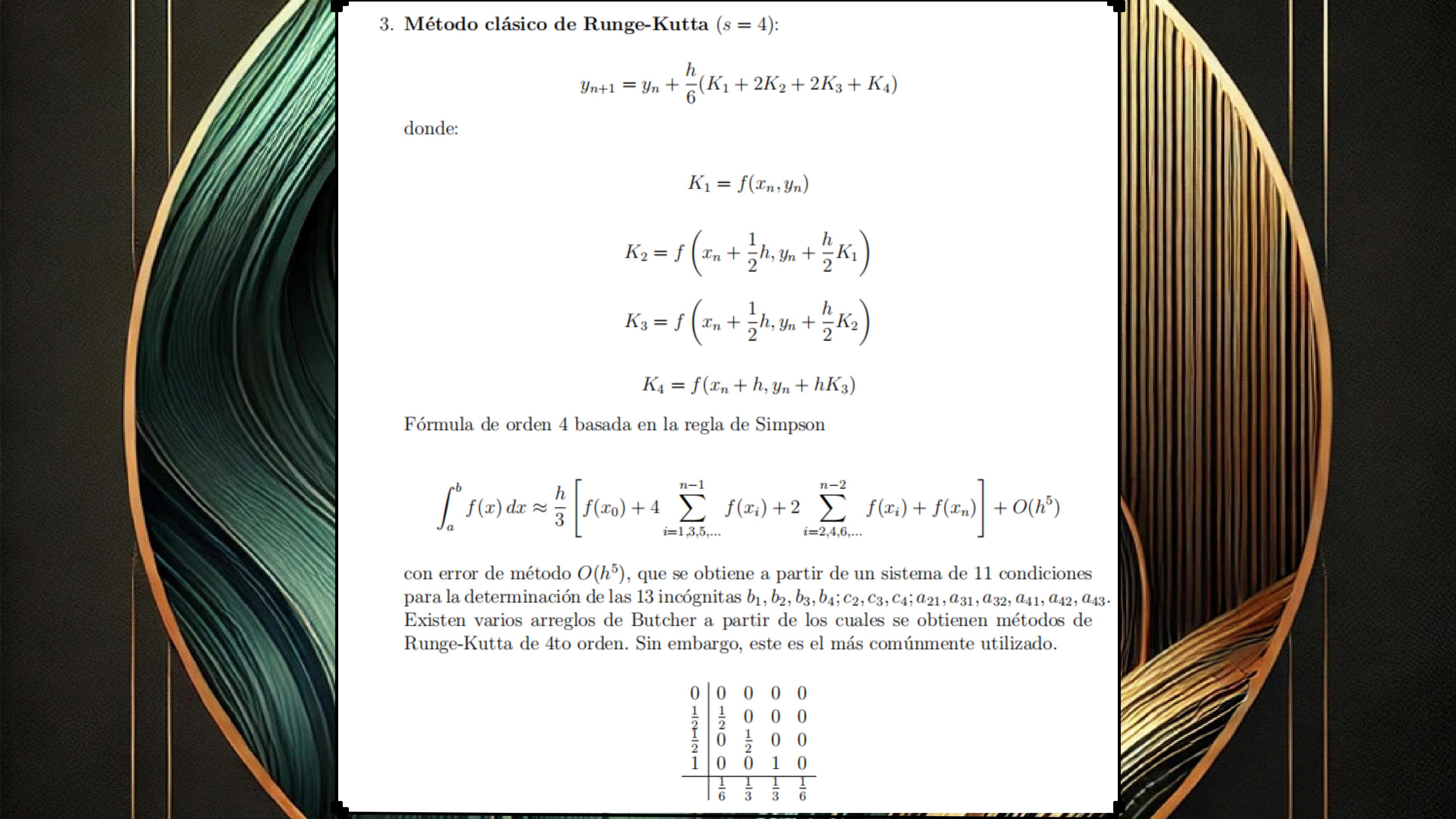
$$y_{n+1} = y_n + h f \left(x_n + \frac{h}{2}, y_n + \frac{h}{2} f(x_n, y_n) \right)$$

Fórmula de orden 2 basada en la regla del punto medio

$$\int_a^b f(x) dx \approx \sum_{i=0}^{n-1} f \left(\frac{x_i + x_{i+1}}{2} \right) \Delta x$$

cuyo arreglo de Butcher sería:

0	0	0
$\frac{1}{2}$	$\frac{1}{2}$	0
	0	1



3. Método clásico de Runge-Kutta ($s = 4$):

$$y_{n+1} = y_n + \frac{h}{6}(K_1 + 2K_2 + 2K_3 + K_4)$$

donde:

$$K_1 = f(x_n, y_n)$$

$$K_2 = f\left(x_n + \frac{1}{2}h, y_n + \frac{h}{2}K_1\right)$$

$$K_3 = f\left(x_n + \frac{1}{2}h, y_n + \frac{h}{2}K_2\right)$$

$$K_4 = f(x_n + h, y_n + hK_3)$$

Fórmula de orden 4 basada en la regla de Simpson

$$\int_a^b f(x) dx \approx \frac{h}{3} \left[f(x_0) + 4 \sum_{i=1,3,5,\dots}^{n-1} f(x_i) + 2 \sum_{i=2,4,6,\dots}^{n-2} f(x_i) + f(x_n) \right] + O(h^5)$$

con error de método $O(h^5)$, que se obtiene a partir de un sistema de 11 condiciones para la determinación de las 13 incógnitas $b_1, b_2, b_3, b_4; c_2, c_3, c_4; a_{21}, a_{31}, a_{32}, a_{41}, a_{42}, a_{43}$. Existen varios arreglos de Butcher a partir de los cuales se obtienen métodos de Runge-Kutta de 4to orden. Sin embargo, este es el más comúnmente utilizado.

0	0	0	0	0
$\frac{1}{2}$	$\frac{1}{2}$	0	0	0
$\frac{1}{2}$	0	$\frac{1}{2}$	0	0
1	0	0	1	0
	$\frac{1}{6}$	$\frac{1}{3}$	$\frac{1}{3}$	$\frac{1}{6}$

Tabla comparativa y conclusiones :

Parámetro	Euler Mejorado	Runge-Kutta (4to orden)
Error de truncamiento o de método	Valor 1	Unidad 1
Error de discretización	Valor 2	Unidad 2
Cantidad de evaluaciones de la función para una misma cantidad de etapas	$2s$	$4s$

Cuadro 1: Comparación entre los métodos de Euler mejorado (EM) y Runge-Kutta de 4to orden (RK4)

En la tabla 1 puede apreciarse que en general RK4 es más preciso que EM, pues para valores pequeños de h ($h < 1$) se cumple que $h^4 < h^2$. Sin embargo, debe valorarse la complejidad de utilizar RK4, puesto que para una misma cantidad de subintervalos necesita realizar el doble de las operaciones (i.e. evaluaciones en la función) que EM; aunque con algo de cuidado puede balancearse el tamaño del paso h con la cantidad de operaciones a realizar para obtener en RK4 una solución más acertada y a la vez más óptima.

Ninguno de los dos métodos anteriormente analizados está exento de errores y debe tenerse especial cuidado en funciones con asíntotas internas que llevan a la función a $\pm\infty$ (o en funciones con cambios drásticos en los valores de su imagen en general), pues los más mínimos errores de cálculo debidos a la aritmética de punto flotante utilizada en el aparato de cómputo sobre el que se trabaja pueden alterar considerablemente los resultados.

Resolución de problemas resueltos

1- Revisión de números famosos

Los siguientes problemas describen los números

$$e \approx 2,71828182846, \quad \ln 2 \approx 0,69314718056, \quad \text{y} \quad \pi \approx 3,14159265359$$

como valores específicos de ciertos problemas de valor inicial. En cada caso, aplicar el método de Runge-Kutta con $n = 10, 20, 40, \dots$ subintervalos (duplicando n en cada ocasión). ¿Cuántos subintervalos son necesarios para obtener, en la sucesión duplicada, el valor correcto del número buscado redondeado a 9 cifras decimales?

1. El número $e = y(1)$, donde $y(x)$ es la solución del problema de valor inicial

$$\frac{dy}{dx} = y, \quad y(0) = 1.$$

2. El número $\ln 2 = y(2)$, donde $y(x)$ es la solución del problema de valor inicial

$$\frac{dy}{dx} = \frac{1}{x}, \quad y(1) = 0.$$

3. El número $\pi = y(1)$, donde $y(x)$ es la solución del problema de valor inicial

$$\frac{dy}{dx} = \frac{4}{1+x^2}, \quad y(0) = 0.$$

Se realizará un acercamiento común a los problemas propuestos a partir de que se desea el valor de cada constante con 9 cifras significativas correctas, o sea, que el valor del error en el método sea menor que 10^{-9} . Como en RK4 el error es $O(h^4)$, podemos hacer $Ch^4 < 10^{-9}$ y calcular el valor de h para el que se cumple la desigualdad, estimando un valor para C (aunque sabemos que depende del comportamiento de la función en el intervalo). Por ejemplo, para $Ch^4 = 10^{-10}$ y $C \leq 1$, vemos que 320 subintervalos es una respuesta válida ($h \approx 0,0031$).

Sin embargo, comprobando en la aplicación desarrollada, vemos que para este valor solo se alcanza una precisión de 7 cifras significativas, e incluso menos en algunos casos, lo cual puede deberse a errores cometidos por aproximaciones necesarias para representar los resultados en la aritmética de punto flotante en la que opera el ordenador. A pesar de que los métodos de Runge-Kutta son absolutamente convergentes, aparentemente no pueden alcanzar el nivel de precisión deseado, ni aumentando el número de subintervalos.

2- Problema 29 página 143

Considere un proyectil disparado verticalmente hacia arriba desde el suelo con una velocidad inicial de 49 m/s. Debido a la resistencia del aire lineal, la función de velocidad $v = \frac{dy}{dt}$ satisface el problema de valor inicial:

$$\frac{dv}{dt} = -0,04v - 9,8, \quad v(0) = 49$$

con solución exacta $v(t) = 294e^{-t/25} - 245$.

- (a) Utilice una implementación del método de Runge-Kutta en una calculadora o computadora para aproximar $v(t)$ en el intervalo $0 \leq t \leq 10$, utilizando tanto $n = 100$ como $n = 200$ subintervalos. Despliegue los resultados en intervalos de 1 segundo. ¿Las dos aproximaciones, cada una redondeada a 4 cifras decimales, coinciden una con otra y con la solución exacta?
- (b) Utilice ahora los datos de la velocidad del inciso (a) para aproximar $y(t)$ en el intervalo $0 \leq t \leq 10$ empleando $n = 200$ subintervalos. Demuestre los resultados en intervalos de 1 segundo. Estos valores de posición aproximados, cada uno redondeado a 2 cifras decimales, coinciden con la solución exacta:

$$y(t) = 7350 (1 - e^{-t/25}) - 245t^2$$

- (c) Si no se contara con la solución exacta, explique cómo se podría usar el método de Runge-Kutta para aproximar cercanamente los tiempos de ascenso y descenso del proyectil y la altura máxima que alcanza.

Utilizando una vía similar a la vista en el problema anterior, acotaremos el error por 10^{-4} . Sustituyendo en la inecuación los valores dados para h (0,1 y 0,05), veremos que para obtener la precisión deseada la constante debe ser menor que 10^4 y 20^4 respectivamente, lo cual es probable. Al comprobar dichas conclusiones en la aplicación, se puede apreciar que