LABORATORIO 5

Unidad 6

Nicolás Delgado
Camila Paladines

Computación Científica

Profesor: Hernán Darío Vargas Cardona, PhD

Junio 11 de 2021

RESUMEN

En este informe se describen los resultados conseguidos al implementar algoritmos en Python para la obtención de respuestas a problemas matemáticos, como lo es el cómputo de la solución de una Ecuación Diferencial Ordinaria (EDO) bajo un Problema de Valor Inicial (PVI) y bajo un Problema de Valor de Frontera (PVF). Las EDOs de primer orden bajo un PVI son calculadas mediante los Métodos de Euler, Serie de Taylor, Runge-Kutta (orden 2 y 4), y Multipaso (2 y 4 pasos). Para las EDOs de orden superior se usa una estrategia de divide y conquista usando los métodos anteriormente mencionados. Las EDOs bajo un PVF son calculadas mediante el Método de Diferencias Finitas y el Método de Elementos Finitos (Colocación). Además, se realizan algunos análisis sobre los métodos utilizados aplicándolos a EDOs de diferente tipo, donde se observa su comportamiento dependiendo de diversos factores. También se calcula la complejidad computacional y la exactitud, con el fin de comparar los métodos y determinar cuál obtuvo mejores resultados para las EDOs usadas.

ABSTRACT

This report describes the results achieved by implementing algorithms in Python to obtain answers to mathematical problems, such as the computation of the solution of an Ordinary Differential Equation (ODE) under an Initial Value Problem (IVP) and under a Boundary Value Problem (BVP). The first-order ODEs under a IVP are calculated using the Euler, Taylor Series, Runge-Kutta (order 2 and 4), and Multi-step (2 and 4 steps) Methods. A divide and conquer strategy using the above mentioned methods is used for the higher-order ODEs. The ODEs under a BVP are calculated using the Finite Difference Method and the Finite Element Method (Collocation). In addition, some analyses are performed on the methods used by applying them to ODEs of different types, where their behavior is observed depending on various factors. The computational complexity and accuracy are also calculated, in order to compare the methods and determine which one obtained better results for the ODEs used.

Contenido

Índice de Figuras

Índice de Tablas

Parte I

Introducción

Parte II
Materiales y Métodos

1. Materiales

Para el desarrollo de esta unidad se usó Python 3.7, con las siguientes librerías:

- `numpy`. Para funciones matemáticas como promedio, desviación estándar, resolución de sistemas de ecuaciones, entre otros.
- `pyplot`. Para graficar los resultados en el plano y las estadísticas de los métodos con respecto a su exactitud y tiempo de ejecución.
- `time`. Para calcular los tiempos de cómputo de cada uno de los métodos en los Problemas de Valor Inicial y los Problemas de Valor de Frontera.
- `sympy`. Para modelar las variables `t` y `y` dentro de las operaciones de los métodos.

2. Métodos

2.1. Problemas de Valor Inicial

Una ecuación diferencial ordinaria (ODE) $y' = f(t, y)$ no tiene una solución única. En general, hay una familia infinita de funciones que satisfacen la ecuación diferencial. Para encontrar una solución particular se debe especificar el valor denotado por y_0 en algún punto t_0. Entonces, se requiere conocer:

$$y(t_0) = y_0$$

Esto se conoce como el problema de valor inicial. La ODE gobierna la evolución dinámica del sistema en el tiempo, desde su valor inicial y_0 en el tiempo t_0. Lo que se busca es una función $y(t)$ que describe el estado de un sistema como función del tiempo.

2.2. ODEs de Primer Orden

La solución analítica de una ODE es una función que se computa en cualquier punto t. Por el contrario, una solución numérica es una tabla de valores aproximados de la solución verdadera evaluados en un conjunto discreto de datos.

Los valores de la solución aproximada se generan paso a paso en incrementos discretos en el tiempo. Por esta razón, los métodos numéricos para resolver ODEs se conocen como métodos de variable discreta.

2.2.1. Método de Euler

Se busca simular el comportamiento del sistema gobernado por la ODE. Empezando en t_0 con el valor inicial dado y_0, se desea seguir la trayectoria dictada por la ODE. La evaluación de $f(t_0, y_0)$ nos dice la pendiente o inclinación de la trayectoria en ese punto. Se usa esta información para predecir el valor y_1 de la solución en el tiempo futuro $t_1 = t_0 + h$, siendo h un incremento adecuado. Considere la siguiente serie de Taylor:

$$y(t+h) = y(t) + y'(t)h + \frac{y''(t)}{2!}h^2 + \cdots +$$

$$= y(t) + f(t, y(t))h + \frac{y''(t)}{2!}h^2 + \cdots +$$

El método de Euler elimina los términos de segundo orden y superior para obtener los valores de la solución aproximada:

$$y_{k+1} = y_k + f(t_k, y_k)h_k$$

Lo cual permite saltar del tiempo t_k a $t_{k+1} = t_k + h_k$. Equivalentemente, si se reemplaza la derivada en la ecuación diferencial $y'(t) = f(t, y)$ por la representación en diferencias finitas, se obtiene una ecuación algebráica:

$$\frac{y_{k+1} - y_k}{h_k} = f(t_k, y_k)$$

2.2.2. Métodos de Serie de Taylor

Del método de Euler se obtiene de la expansión en series de Taylor. Si se toman más términos en la serie de Taylor, se pueden obtener más métodos de paso sencillo (*single-step*) para órdenes más altos. Por ejemplo, tomando un término adicional en la serie de Taylor:

$$y(t+h) = y(t) + y'(t)h + \frac{y''(t)}{2!}h^2 + \frac{y'''(t)}{3!}h^3 + \cdots +$$

Se obtiene el método de segundo orden:

$$y_{k+1} = y_k + y_k^{'}h_k + \frac{y_k^{''}}{2}h_k^2$$

Nótese que se requiere la segunda derivada de y''. Esto se puede obtener diferenciando $y' = f(t, y)$ usando la regla de la cadena:

$$y'' = f_t(t, y) + f_y(t, y)y' = f_t(t, y) + f_y(t, y)f(t, y)$$

2.2.3. Métodos de Runge-Kutta

Son métodos de paso sencillo que son similares a los métodos de serie de Taylor, pero no requieren la computación de derivadas de orden superior. En vez de eso, se simula el efecto de las derivadas superiores evaluando f varias veces entre t_k y t_{k+1}. La idea básica se ilustra por el método de *Runge-Kutta* de orden 2, conocido como el método de *Heun*.

$$f_t + f_{y_f} = \frac{f(t+h, y+h_f) - f(t, y)}{h}$$

Con esta aproximación, el método de Taylor de segundo orden se convierte a (el cual es de **exactitud de orden 2**):

$$y_{k+1} = y_k + \frac{f(t_k, y_k) + f(t_k + h_k, y_k + h_{k_f}(t_k, y_k))}{2}h_k$$

El método de Runge-Kutta mejor conocido y más empleado es el ***esquema clásico de orden 4***:

$$y_{k+1} = y_k + \frac{1}{6}\left(k_1 + 2k_2 + 2k_3 + k_4\right),$$

donde:

- $k_1 = f(t_k, y_k)h_k$,
- $k_2 = f(t_k + \frac{h_k}{2}, y_k + \frac{k_1}{2})h_k$,
- $k_3 = f(t_k + \frac{h_k}{2}, y_k + \frac{k_2}{2})h_k$,
- $k_4 = f(t_k + h_k, y_k + k_3)h_k$.

2.2.4. Métodos Multi-Paso

Estos métodos usan informacién en más de un punto previo para estimar la solución en el siguiente. Por esta razón, a veces se conocen como métodos con memoria. Los métodos lineales *multi-paso* tienen la siguiente forma general:

$$y_{k+1} = \sum_{i=1}^{n} \alpha_i y_{k+1-i} + h \sum_{i=0}^{n} \beta_i f(t_{k+1}, y_{k+1-i})$$

Los parámetros α_i y β_i se determinan con interpolación polinómica. Si $\beta_0 = 0$, el método es explícito, de lo contrario es implícito. La fórmula explícita del método de dos pasos es:

$$y_{k+1} = y_k + \frac{1}{2}(3y_k^{'} - y_{k-1}^{'})h$$

Uno de los métodos multi-paso más populares es el método explícito de cuarto orden de *Adams-Bashforth*:

$$y_{k+1} = y_k + \frac{1}{24}\left(55y_k^{'} - 59y_{k-1}^{'} + 37y_{k-2}^{'} - 9y_{k-3}^{'}\right)h$$

2.3. ODEs de Orden Superior

Las ecuaciones diferenciales de orden superior ocurren frecuentemente en la práctica, pero pueden ser transformadas a un sistema equivalente de primer orden de la siguiente manera:

$$u^{(n)} = f(t, u, u', \ldots, u^{(n-1)})$$

La ecuación de orden n-ésimo define las n nuevas incógnitas $y_1(t) = u$, $y_2(t) = u', \ldots, y_n(t) = u^{(n-1)}$, tal que la ecuación original se convierte en un sistema de primer orden con n ecuaciones:

$$\begin{bmatrix} y_1' \\ y_2' \\ \vdots \\ y_{n-1}' \\ y_n' \end{bmatrix} = \begin{bmatrix} y_2 \\ y_3 \\ \vdots \\ y_n \\ f(t, u, u', \ldots, u^{(n-1)}) \end{bmatrix}$$

2.4. Problemas de Valor de Frontera para ODEs

Un problema de valor de frontera (PVF) para una ODE especifica más de un punto en el cual la solución o sus derivadas deben tener valores dados. Por ejemplo, un PVF para una ODE de segundo orden tiene la forma:

$$y'' = f(t, y, y'),\ a \leq t \leq b,$$

con condiciones de frontera:

$$y(a) =,\ y(b) = \beta$$

En general, para hallar la solución particular, deben existir tantas condiciones como el orden de la ODE.

Si todas las condiciones se especifican en un mismo punto, entonces tenemos un problema de valor inicial (PVI), de lo contrario tenemos un (PVF).

Por ejemplo, un PVF para un sistema de dos ODE de primer orden tiene la forma:

$$\begin{bmatrix} y_1' \\ y_2' \end{bmatrix} = \begin{bmatrix} f_1(t,y) \\ f_2(t,y) \end{bmatrix}, \ a \leq t \leq b$$

Con C.F: $y_1(a) =$, $y_1(b) = \beta$.

2.4.1. Método de Diferencias Finitas

El método de diferencias finitas convierte los PVF en sistemas de ecuaciones algebraicas, reemplazando algunas derivadas por aproximaciones con diferencias finitas. Por ejemplo, para resolver un PVF de dos puntos:

$$y'' = f(t, y, y'), \ a \leq t \leq b,$$

con CF: $y(a) =, y(b) = \beta$. Primero se divide el intervalo $[a, b]$ en n subintervalos igualmente espaciados. Sea $t_i = a + ih$, $i =$ 0, 1, 2, $\ldots$, n, donde $h = \frac{b-a}{n}$. Se busca una aproximación $y_i \approx y(t_i)$ en cada uno de los puntos de la malla datos por t_i.

Ya tenemos $y_0 =$ y $y_n = \beta$. Ahora reemplazamos las derivadas con aproximaciones de diferencias finitas:

$$y'(t_i) \approx \frac{y_{i+1} - y_{i-1}}{2h}, \ y''(t_i) \approx \frac{y_{i+1} - 2y_i + y_{i-1}}{h^2}.$$

Este reemplazo lleva al siguiente sistema de ecuaciones algebraicas:

$$y_{i+1} - 2y_i + y_{i-1} - h^2 f(t_i, y_i, \frac{y_{i+1} - y_{i-1}}{2h}) = 0$$

El cual debe ser resuelto para las incógnitas y_i, $i = 1, \ldots, n-1$. El sistema de ecuaciones puede ser lineal o no, dependiendo de que f sea lineal en y y y'.

2.4.2. Método de Elementos Finitos

Este método aproxima la solución de un PVF a una combinación lineal de funciones base ϕ_i, típicamente monomios. La aproximación tiene la forma:

$$y(t) \approx u(t) = \sum_{i=1}^{n} x_i \phi_i(t)$$

Los coeficientes x_i están determinados mediante la imposición de requerimientos del residuo, que se define como la diferencia entre el lado izquierdo y derecho de la ODE. En este laboratorio se usa el Método de Colocación en la que el residuo es cero, y la ODE se satisface exactamente en los n puntos discretos.

Parte III
Resultados de las Simulaciones

3. ODEs de Primer Orden

3.1. Ejemplo 1

3.2. Ejemplo 2

3.3. Ejemplo 3

4. ODEs de Orden Superior

5. Problemas de Valor de Frontera

Parte IV

Discusión y Análisis

6. ODEs de Primer Orden

7. ODEs de Orden Superior

8. Problemas de Valor de Frontera

Parte V
Conclusiones

Parte VI

Referencias

- Material del curso, disponible en BlackBoard
- Bornemann, F., 2016. *Numerical linear algebra.* 1st ed. *Simson, W.*
- Mathews, J., Fink, K., Fernández Carrión, A. & Contreras Márquez, M., 2011. *Métodos Numéricos con MATLAB*. 3rd ed. Madrid: *Pearson Prentice Hall.*
- Librería Numpy
- Librería Pyplot (Matplotlib)
- Librería Time
- Librería Sympy