

# Análisis Numérico de Ecuaciones Diferenciales Ordinarias

## Tarea II

Daniel Castañón Quiroz<sup>\*1</sup>

<sup>1</sup>Departamento de Matemáticas y Mecánica, IIMAS-UNAM, Cd. de México, México

September 10, 2025

### 1 Instrucciones generales

Todo los problemas se deberán entregar en archivos diferentes con extensión `.m`. Por ejemplo el problema 1 deberá estar en el archivo `Problema_1.m`, etc.

Dentro de cada archivo se deberá poner el nombre del estudiante y su correo electrónico. Utiliza comentarios cuando sea necesario. Todos los programas deberán correr y tener solamente el output que se especifica utilizando el comando `disp` y/o la(s) gráfica(s) que se pide(n) utilizando el comando `plot`.

### 2 Problemas en Matlab

#### 2.1 Problemas de resolución numérica de sistemas no-lineales en Matlab

- 1.- (1 punto) Utilizar el método de Newton con criterio de tolerancia de paro de  $\epsilon = 10^{-9}$  de la diferencia absoluta entre aproximaciones sucesivas, y un máximo de iteraciones  $N = 15$  para aproximar la raíz positiva de la ecuación  $g(x) = 0$ , donde  $g(x) := x^2 - 5$ . Escoger como aproximación inicial  $x_0 = 1$ . Al final del algoritmo el programa deberá imprimir la solución aproximada  $\hat{x}$ , el valor  $g(\hat{x})$  y la gráfica de la evolución de  $|g(x_n)|$  confirmando que efectivamente ésta es cuadrática.
- 2.- (1 punto) Utilizar el método de Newton con criterio de tolerancia de paro de  $\epsilon = 10^{-9}$  de la diferencia en la norma euclideana entre aproximaciones sucesivas, y un máximo de iteraciones  $N = 15$  para aproximar la solución al sistema no-lineal  $\mathbf{g}(\mathbf{x}) = \mathbf{0}$ , donde  $\mathbf{g}(\mathbf{x}) = [g_1(\mathbf{x}) \ g_2(\mathbf{x}) \ g_3(\mathbf{x})]^t$  y

$$g_1(\mathbf{x}) = 3x_1 - \cos(x_1x_2) - \frac{1}{2}, \quad (1)$$

$$g_2(\mathbf{x}) = x_1^2 - 81(x_2 + 0.1)^2 + \sin x_3 + 1.06, \quad (2)$$

$$g_3(\mathbf{x}) = e^{-x_1x_2} + 20x_3 + \frac{10\pi - 3}{3}. \quad (3)$$

Escoger como aproximación inicial  $\mathbf{x}_0 = [0.1 \ 0.1 \ -0.1]^t$ . Al final del algoritmo el programa deberá imprimir la solución aproximada  $\hat{\mathbf{x}}$ , la norma euclideana de  $\mathbf{g}(\hat{\mathbf{x}})$ , y la gráfica de la evolución de  $\|\mathbf{g}(\mathbf{x}_n)\|$  confirmando que efectivamente ésta es cuadrática.

---

<sup>\*</sup>[daniel.castanon@iimas.unam.mx](mailto:daniel.castanon@iimas.unam.mx)

## 2.2 Instrucciones para los Problemas 3, 4 y 5

En estos tres problemas se pide programar un método numérico en particular para un problema en particular. Adicionalmente, el programa debe ser capaz de obtener el máximo del error absoluto en la norma euclídeana y la tasa de convergencia del método. Para ello utilizar un número de subintervalos de  $N = 4, 8, 16, 32, 64, 128$ . Se recomienda tomar como partida de referencia el programa **Matlab #5** que esta en el website del curso. El único output del programa deber ser una tabla de la forma:

[N\_vec' err\_max' err\_rate']

donde N\_vec es el número de subintervalos que divide al interval  $[0, 1]$ , err\_max el error máximo absoluto en la norma euclídeana, y err\_rate la tasa de convergencia.

3.- (1 punto) Sea el problema de valor inicial en el intervalo  $0 \leq t \leq 1$ :

$$\begin{aligned}u_1' &= u_1 + 2u_2 - 2u_3 + e^{-t}, \\u_2' &= u_2 + u_3 - 2e^{-t}, \\u_3' &= u_1 + 2u_2 + e^{-t},\end{aligned}$$

sujeta a  $u_1(0) = 3, u_2(0) = -1, u_3(0) = 1$ ; su solución analítica es

$$\begin{aligned}u_1(t) &= -3e^{-t} - 3 \sin(t) + 6 \cos(t), \\u_2(t) &= \frac{3}{2}e^{-t} + \frac{3}{10} \sin(t) - \frac{21}{10} \cos(t) - \frac{2}{5}e^{2t}, \\u_3(t) &= -e^{-t} + \frac{9}{5} \sin(t) + \frac{12}{5} \cos(t) - \frac{2}{5}e^{2t}.\end{aligned}\tag{4}$$

Programar el método del punto de medio para sistemas (véase problema 2 de la tarea I) para resolver este problema. Recordar que el método del punto medio tiene un error local de truncamiento/consistencia de  $O(\Delta t^2)$ . Entonces para confirmar que el método es implementado correctamente, verificar (utilizando las instrucciones mencionadas al inicio de la Sección 2.2) que la tasa de convergencia calculada debe ser cercana a 2.

4.- (1 punto) Sea el problema de valor inicial en el intervalo  $0 \leq t \leq 1$ :

$$\mathbf{u}' = A\mathbf{u} + \mathbf{g}(t),$$

donde  $\mathbf{u}$  es una función de  $\mathbb{R} \rightarrow \mathbb{R}^3$ , y

$$A = \begin{bmatrix} 1 & 2 & -2 \\ 0 & 1 & 1 \\ 1 & 2 & 0 \end{bmatrix}, \quad \mathbf{g}(t) = \begin{bmatrix} e^{-t} \\ -2e^{-t} \\ e^{-t} \end{bmatrix}.$$

Este problema tiene como condición inicial  $\mathbf{u}(0) = [3 \quad -1 \quad 1]^t$ . Obsérvese que este problema es el mismo que el problema 1, por lo tanto tiene como solución analítica la ecuación (4). Programar el método de **Taylor implícito** para sistemas de orden  $m = 2$  definido por:

$$\begin{aligned}\hat{\mathbf{u}}_0 &= \mathbf{u}_0 \\ \hat{\mathbf{u}}_{i+1} &= \hat{\mathbf{u}}_i + \Delta t \mathbf{F}(t_{i+1}, \hat{\mathbf{u}}_{i+1}) - \frac{1}{2} \Delta t^2 \frac{d}{dt} [\mathbf{F}(t_{i+1}, \hat{\mathbf{u}}_{i+1})] \quad \text{para } i \in \{0, \dots, N\},\end{aligned}$$

donde  $\mathbf{F}(t, \mathbf{u}) = A\mathbf{u} + \mathbf{g}(t)$ . Dado que este método es implícito y  $\mathbf{F}(t, \mathbf{u})$  es lineal con respecto a  $\mathbf{u}$ , en cada iteración se debe resolver un sistema lineal de  $3 \times 3$ . Para ello utilizar el comando

de `linsolve` o el operador de división izquierda ‘\’ de Matlab (véase Libro de Gilat-Matlab página 71). Este método tiene un error local de truncamiento/consistencia de  $O(\Delta t^2)$ , por lo tanto verificar que la tasa de convergencia calculada debe ser cercana a 2.

**Medio punto extra:** Probar formalmente que el error local de truncamiento/consistencia de este método de Taylor implícito para sistemas es de orden 2 (adjuntar esta prueba en .pdf ya sea generado por latex o hacerlo a mano y escanearlo).

- 5.- (1 punto) Sea el problema de valor inicial en el intervalo  $0 \leq t \leq 2$ :

$$y' = -(y + 1)(y + 3),$$

sujeta a  $y(0) = -2$ ; su solución analítica es

$$y(t) = -3 + \frac{2}{1 + e^{-2t}}.$$

Programar el método del trapecio para resolver este problema. Utilizar el método de Newton con criterio de paro de  $\epsilon = 10^{-9}$  de la diferencia absoluta entre aproximaciones sucesivas, y un máximo de iteraciones  $N_0 = 15$  en cada iteration en el tiempo. Recordar que este método tiene un error local de truncamiento/consistencia de  $O(\Delta t^2)$ , entonces verificar que la tasa de convergencia calculada debe ser cercana a 2.

- 6.- (2 puntos) Programar en Matlab (utilizando el paquete ‘Symbolic toolbox’) el método de **Picard** para resolver el ejemplo 1 de la página 70 del libro de Braun y el ejercicio 1 de la página 80. En ambos problemas dar suficiente evidencia ‘algebraica y visual (graficar)’ de que los cálculos efectivamente convergen a la solución.

### 2.3 Instrucciones para los Problemas 7 y 8

En estos problemas el único output deber ser una gráfica utilizando el comando `plot`.

- 7.- (1 punto) Resolver el problema del **péndulo simple**, el cual es modelado por el problema de condición inicial:

$$\theta'' + \frac{g}{L} \sin(\theta) = 0, \quad (5)$$

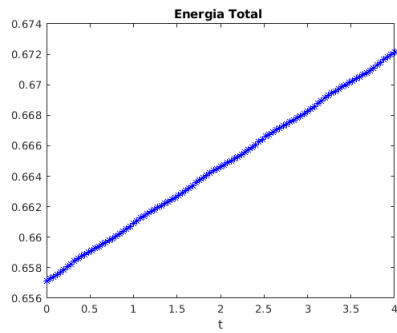
donde  $\theta(t)$  esta sujeta a  $\theta(0) = \frac{\pi}{6}$  y  $\theta'(0) = 0$ . Sean  $g = 9.81 \text{m/s}^2$  y  $L = 0.5 \text{m}$ , entonces realizar lo siguiente:

- Definir  $\theta_1(t) := \theta$  y  $\theta_2(t) := \theta'$  y plantear el problema (5) como un sistema no-lineal de EDOs.
- Resolver el sistema utilizando el método del punto medio para  $t \in [0, 4]$ , y un número de sub-intervalos  $N = 100$ .
- Utilizando la aproximaciones obtenidas de  $\theta$  y su derivada  $\theta'$ , graficar la energía total del sistema definida por:

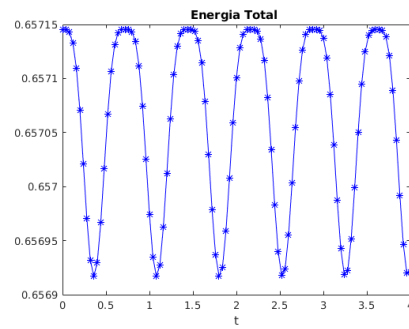
$$E(t) = \frac{1}{2}(L\theta')^2 + gL(1 - \cos(\theta)).$$

La forma de la gráfica obtenida debe ser similar a la que se muestra en la Figura 1a.

- 8.- (2 puntos) Resolver todos los incisos del problema anterior utilizando el método del trapecio con el método de Newton con criterio de paro de  $\epsilon = 10^{-9}$  de la diferencia absoluta entre aproximaciones sucesivas, y un máximo de iteraciones  $N_0 = 15$ . La forma de la gráfica de la energía total obtenida debe ser similar a la que se muestra en la Figura 1b.



(a) Utilizando el método del punto medio.



(b) Utilizando el método del trapecio con Newton.

Figure 1: Energía total para el problema del péndulo simple.