Лабораторна робота 5
Чисельні методи розв'язання задачі Коші для звичайних диференціальних рівнянь

5.1. Мета роботи

Вивчення однокрокових і багатокрокових чисельних методів для практичного розв'язання задачі Коші (задачі з початковою умовою) для звичайних диференціальних рівнянь, придбання навичок використання цих методів для розв'язання задачі Коші із застосуванням комп'ютера.

5.2. Методичні рекомендації щодо організації самостійної роботи

За темою лабораторної роботи студент повинен: *знати* загальне формулювання задачі Коші (задачі з початковою умовою) для звичайних диференціальних рівнянь; *вміти* розв'язувати цю задачу з використанням однокрокових і багатокрокових чисельних методів [1, 2, 6].

Задача Коші (задача з початковою умовою) для звичайного диференціального рівняння виду

$$y' = f(x, y) \tag{5.1}$$

полягає в такому: знайти функцію $y = y(x)$ на заданому відрізку $[a, b]$, що задовольняє рівнянню (5.1) і початковій умові

$$y(a) = y_0, \tag{5.2}$$

де y_0 задано.

Чисельні методи розв'язання задачі (5.1) – (5.2) знаходять розв'язок (тобто функцію $y(x)$ на відрізку $[a, b]$) у табличному вигляді, а саме у вигляді набору точок (x_i, y_i), $i = \overline{0, n}$, де $x_0 = a$, $x_i = x_0 + i \times h$, $i = \overline{1, n}$, $h = \dfrac{b-a}{n}$, n – задане число розбиття відрізка $[a, b]$, y_i, $i = \overline{1, n}$ – знайдені наближені значення функції $y(x)$ у вузлах сітки x_i.

Розглянемо кілька методів розв'язання задачі Коші для диференціального рівняння (5.1) на відрізку $[a, b]$.

5.2.1. Однокрокові методи

Метод Ейлера. Значення y_i обчислюються рекурентно за формулою:

$$y_{i+1} = y_i + h \times f(x_i, y_i),\ i = \overline{0, n-1}.$$

Похибка методу Ейлера, оцінювана для величини $|y_n - y(x_n)|$, має порядок $O(h)$ [1 – 3; 5; 6].

1-ша модифікація методу Ейлера. Значення y_i обчислюються рекурентно за формулами:

$$y_{i+\frac{1}{2}} = y_i + \frac{h}{2} f(x_i, y_i),\ y_{i+1} = y_i + h \times f(x_i + \frac{h}{2}, y_{i+\frac{1}{2}}),\ i = \overline{0, n-1}.$$

Похибка 1-ої модифікації методу Ейлера, оцінювана для величини $|y_n - y(x_n)|$, має порядок $O(h^3)$ [1 – 3; 5; 6].

2-га модифікація методу Ейлера. Значення y_i обчислюються рекурентно за формулами:

$$y^*_{i+1} = y_i + h \times f(x_i, y_i),\ y_{i+1} = y_i + \frac{h}{2}(f(x_i, y_i) + f(x_{i+1}, y^*_{i+1})),\ i = \overline{0, n-1}.$$

Похибка 2-ої модифікації методу Ейлера, оцінювана для величини $|y_n - y(x_n)|$, має порядок $O(h^3)$ [1 – 3; 5; 6].

Метод Рунге – Кутта 4-го порядку. Значення y_i обчислюються рекурентно за формулами:

$$y_{i+1} = y_i + \frac{1}{6}(k_1 + 2k_2 + 2k_3 + k_4),\ i = \overline{0, n-1},$$

де $k_1 = h \times f(x_i, y_i)$, $k_2 = h \times f(x_i + \frac{h}{2}, y_i + \frac{k_1}{2})$, $k_3 = h \times f(x_i + \frac{h}{2}, y_i + \frac{k_2}{2})$, $k_4 = h \times f(x_i + h, y_i + k_3)$.

Похибка методу Рунге – Кутта 4-го порядку, оцінювана для величини $|y_n - y(x_n)|$, має порядок $O(h^4)$ [1 – 3; 5; 6].

5.2.2. Багатокрокові методи

У **методі прогнозу та корекції** 4-го порядку при розв'язанні задачі Коші (5.1) значення y_i у вузлах сітки x_i, $i = \overline{0, n}$, обчислюються рекурентно за формулами:

$$y^{(pred)}_{i+1} = y_i + \frac{h}{24}(-9f_{i-3} + 37f_{i-2} - 59f_{i-1} + 55f_i),$$

$$f_{i+1}^{(pred)} = f(x_{i+1}, y_{i+1}^{(pred)}),$$

$$y_{i+1} = y_i + \frac{h}{24}(f_{i-2} - 5f_{i-1} + 19f_i + 9f_{i+1}^{(pred)}),$$

де $f_i = f(x_i, y_i)$.

До початку розрахунків за методом прогнозу та корекції 4-го порядку необхідно 3 перші кроки зробити будь-яким однокроковим методом, наприклад, методом Рунге – Кутта.

Похибка методу прогнозу та корекції 4-го порядку, оцінювана для величини $\|y_n - y(x_n)\|$, має порядок $O(h^4)$ [1; 2; 6].

5.3. Контрольні приклади

Приклад 1. Розв'язати методом Ейлера задачу Коші: $y' = y - \frac{2x}{y}$, $y(0) = 1$ на відрізку [0,1] с числом розбиття відрізка $n = 20$.

Розв'язання. Складемо процедуру для метода Ейлера:

```
def Eiler(FunPr, a, b, y0, n):
  h  = (b - a)/n
  x = [0]*n
  y = [0]*n
  x[0] = a
  y[0] = y0
  for i in range(n-1):
    x[i+1] = x[i] + h
    y[i+1] = y[i] + h*FunPr(x[i], y[i])
  return [x, y]
```

Вводимо дані для розв'язання задачі:

```
# Задаємо рівняння (функцію правої частини)
def MyFunPr(x, y):
  return y - 2*x/y

a = 0
b = 1
y0 = 1
n = 20
```

Викликаємо записану процедуру і отримуємо розв'язок диференціального рівняння у вигляді таблично представленої функції:

```
[X, Y] = Eiler(MyFunPr, a, b, y0, n)
print(X)
print(Y)
```

```
[0, 0.05, 0.1, 0.15000000000000002, 0.2, 0.25, 0.3, 0.35, 0.39999999999999997, 0.44999999999999996,
[1, 1.05, 1.0977380952380953, 1.143515358963236, 1.1875736817608582, 1.230111305536462, 1.271293506
```

Для порівняння також отримаємо розв'язок за допомогою процедури *solve_ivp* пакету *scipy.integrate* для Python*:*

```
from scipy.integrate import solve_ivp

t_eval = np.arange(a, b, (b-a)/n)
sol = solve_ivp(MyFunPr, [a, b], [y0], t_eval=t_eval)
sol
```

```
  message: The solver successfully reached the end of the integration interval.
  success: True
   status: 0
        t: [ 0.000e+00  5.000e-02 ...  9.000e-01  9.500e-01]
        y: [[ 1.000e+00  1.049e+00 ...  1.674e+00  1.703e+00]]
      sol: None
 t_events: None
 y_events: None
     nfev: 20
     njev: 0
      nlu: 0
```

Будуємо графіки отриманих розв'язків:

```
import matplotlib.pyplot as plt

plt.figure(figsize=(5,5))
plt.plot(X, Y)
plt.plot(sol.t, sol.y.T[:, 0], color="red")
```

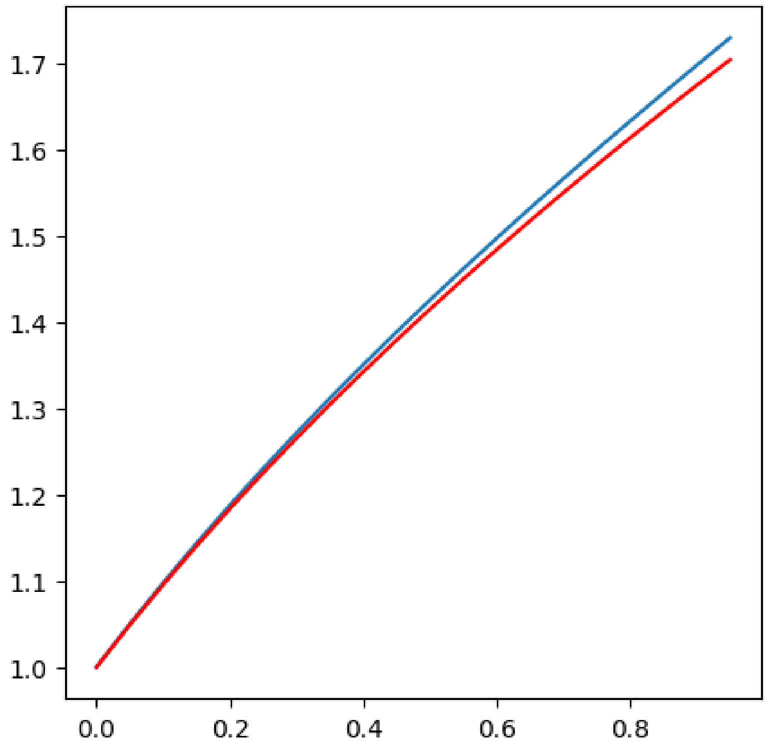

Рис. 5.1. Графічне представлення розв'язку диференціального рівняння

Як бачимо з графіків, при застосуванні різних методів розв'язання задачі розв'язки різняться і ця різниця збільшується до правого краю відрізка *[a, b]*.

Приклад 2. Процес руху автомобіля на площині в найпростішому випадку може бути описаний системою диференціальних рівнянь [16]:

$$\begin{cases} \dfrac{\partial x}{\partial t} = V\cos\theta \\ \dfrac{\partial y}{\partial t} = V\sin\theta \\ \dfrac{\partial \theta}{\partial t} = -\dfrac{V}{W\,Ctg\phi + \dfrac{w}{2}} \end{cases}, \qquad (5.3)$$

де (x, y) – координати точки M на площині xOy, θ – кут між повздовжньою віссю автомобіля й віссю Ox, V – швидкість, ϕ – кут повороту передніх коліс відносно повздовжньої осі автомобіля, W – відстань між передньою і задньою осями (колісна база), w – відстань між колесами автомобіля на задній осі (колія задніх коліс) (рис. 5.2). У моделі (5.3) всі кути вимірюються в радіанах.

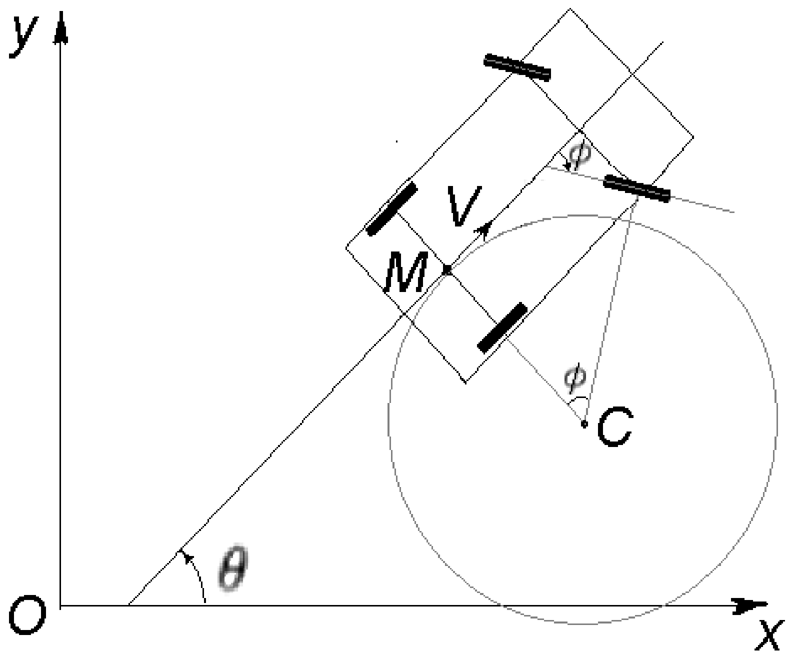

Рис. 5.2. Модель руху автомобіля на площині

Треба визначити траєкторію руху автомобіля (тобто точки M) впродовж 6 секунд, якщо його швидкість V була постійна і дорівнювала 5 км/год., кут ϕ також був постійним і дорівнював 30^0, $W = 2.47$ м, $w = 1.456$ м, на початку руху точка M мала координати $(5;\ 2)$, а кут $\theta = 0$.

Розв'язання. Оскільки в даному прикладі маємо три координати (x, y, θ), які змінюються у часі, а також три диференціальних рівняння в системі (5.3); то процедуру для метода Ейлера перепишемо в такому вигляді:

```
import numpy as np

def MetEiler(FunPr, a, b, Y0, n):
  h  = (b - a)/n
  t = np.zeros((n))
  Y = np.zeros((n, 3))
  t[0] = a
  Y[0,:] = Y0
  for i in range(n-1):
    t[i+1] = t[i] + h
    Y[i+1,:] = Y[i,:] + h*FunPr(t[i], Y[i,:])
  return [t, Y]
```

Вводимо дані для розв'язання задачі:

```
import math

W = 2.7
w = 1.456
def V(t):
  return 5
def Fi(t):
  return math.pi/6

# Задаємо рівняння (функцію правої частини)
def FunPr(t, Y):
  Yt = np.zeros((3))
  Yt[0] = V(t)*math.cos(Y[2])
  Yt[1] = V(t)*math.sin(Y[2])
  Yt[2] = -V(t)/(W/math.tan(Fi(t)) + w/2)
  return Yt

a = 0
b = 6
Y0 = [5, 2, Fi(a)]
n = 20
```

Викликаємо записану процедуру і отримуємо розв'язок:

```
[t, Y] = MetEiler(FunPr, a, b, Y0, n)
```

Для порівняння також отримаємо розв'язок за допомогою процедури *solve_ivp* пакету *scipy.integrate* для Python:

```python
from scipy.integrate import solve_ivp

t_eval = np.arange(a, b, (b-a)/n)
sol = solve_ivp(FunPr, [a, b], Y0, t_eval=t_eval)
```

Будуємо графіки отриманих розв'язків:

```python
import matplotlib.pyplot as plt

plt.figure(figsize=(10, 10))
plt.scatter(Y0[0], Y0[1], color = "green")
plt.plot(Y[:, 0], Y[:, 1])
plt.scatter(Y[-1, 0], Y[-1, 1], color = "blue")
plt.plot(sol.y.T[:, 0], sol.y.T[:, 1], color="red")
plt.scatter(sol.y.T[-1, 0], sol.y.T[-1, 1], color = "red")
```

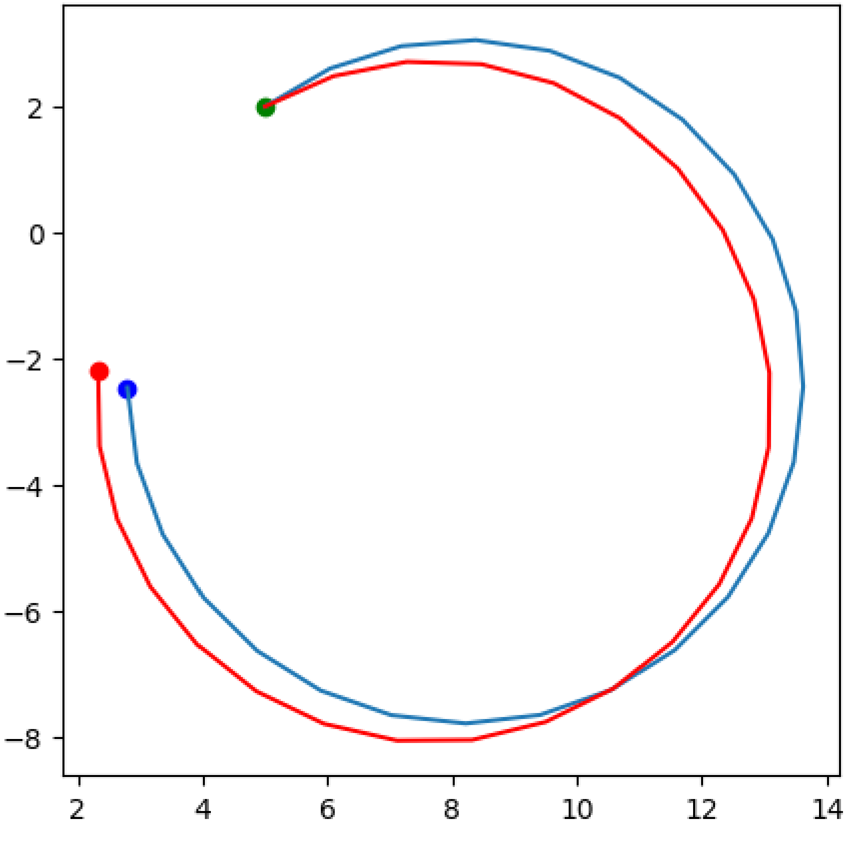

Рис. 5.3. Траєкторія руху автомобіля

Як бачимо, при постійному куті повороту передніх коліс рух автомобіля буде відбуватись по колу (анімація  lab_05.mp4). Також бачимо, що при застосуванні різних методів розв’язання задачі розрахункова траєкторія руху автомобіля різна.

Приклад 3. Припустимо, що рух гоночного автомобіля описується рівнянням $y' = \frac{1.3y}{t}$, де y – відстань, пройдена за час t з моменту початку руху.

Нехай через півгодини автомобіль віддалився на 75 км. Необхідно визначити на яку відстань він від'їде через 4 години після початку руху.

*Розв'язання***.** Для розв'язання скористаємося записаною в прикладі 1 процедурою E*iler*:

```
# Задаємо рівняння (функцію правої частини)
def MyFunPr(t, y):
  return 1.3*y/t

a = 0.5
b = 4
y0 = 75
n = 100

[X,Y] = Eiler(MyFunPr, a, b, y0, n)

print(Y[-1])
```

```
1094.0643509175866
```

Таким чином, за 4 години гоночний автомобіль подолає відстань 1 094,064 км.

Для порівняння також отримаємо розв'язок за допомогою процедури *solve_ivp* пакету *scipy.integrate* для Python*:*

```
from scipy.integrate import solve_ivp

t_eval = np.arange(a, b, (b-a)/(n+1))
sol = solve_ivp(MyFunPr, [a, b], [y0], t_eval=t_eval)
print(sol.y.T[-1, 0])
```

```
1106.977913499793
```

Як бачимо, при застосуванні різних методів розв’язання задачі розв’язки різні.

5.4. Порядок виконання роботи і варіанти завдань

5.4.1. Зміст звіту

У теоретичній частині роботи необхідно коротко описати:

– постановку задачі Коші для звичайного диференціального рівняння 1-го порядку;

– чисельні однокрокові методи розв'язання задачі Коші для звичайного диференціального рівняння 1-го порядку;

– метод прогнозу та корекції розв'язання задачі Коші для звичайного диференціального рівняння 1-го порядку.

У практичній частині роботи необхідно:

– запрограмувати у вигляді окремих модулів метод Ейлера, 1-шу і 2-гу модифікації методу Ейлера та метод Рунге – Кутта 4-го порядку для розв'язання задачі Коші для рівняння (5.1);

– запрограмувати у вигляді окремого модуля алгоритм за яким можна розв'язати задачу Коші з заданою точністю;

– запрограмувати у вигляді окремого модуля метод прогнозу та корекції для розв'язання задачі Коші для рівняння (5.1);

– привести тексти складених програм;

– розв'язати задачу Коші для диференціального рівняння (5.1) чисельно методами Ейлера і його модифікаціями й методом Рунге – Кутта із числом розбиття відрізка $n = 20$;

– розв'язати задачу Коші для диференціального рівняння (5.1) чисельно методом прогнозу та корекції із числом розбиття відрізка $n = 20$;

– розв'язати задачу Коші для диференціального рівняння (5.1) за допомогою процедури *solve_ivp* пакету *scipy.integrate* із числом розбиття відрізка $n = 20$;

– побудувати графіки отриманих розв'язків (для порівняння в одному графічному вікні);

– розв'язати задачу Коші для диференціального рівняння (5.1) чисельно методом Ейлера с точністю $\varepsilon = 10^{-7}$.

5.4.2. Варіанти індивідуальних завдань

Варіант 1. $y' = x^2 + y^2, \quad [0,1], \quad y(0) = 0$.

Варіант 2. $y' = 1 + xy^2, \quad [0,1], \quad y(0) = 0$.

Варіант 3. $y' = \dfrac{y}{x+1} - y^2, \quad [0,1], \quad y(0) = 1$.

Варіант 4. $y' = \dfrac{y}{x}, \quad [1,25], \quad y(1) = 1$.

Варіант 5. $y' = \dfrac{x^2 - y^2}{xy}, \quad [2,25], \quad y(2) = 1$.

Варіант 6. $y' = \frac{1}{2}xy$, $[0,1]$, $y(0) = 1$.

Варіант 7. $y' = \frac{xy}{x^2 - y^2}$, $[2,25]$, $y(2) = 1$.

Варіант 8. $y' = \frac{x + y - 3}{x - y - 1}$, $[2,25]$, $y(2) = 1$.

Варіант 9. $y' = \frac{2xy - 1}{4x + 2y + 5}$, $[2,25]$, $y(2) = 1$.

Варіант 10. $y' = \frac{3x - y + 2}{2x - 3y + 1}$, $[3,30]$, $y(3) = 5$.

Варіант 11. $y' = \frac{xy^3}{x^2 + y^2}$, $[5,25]$ $y(5) = 7$.

Варіант 12. $y' = \frac{y^4 + y}{x^2}$, $[2,34]$ $y(2) = 10$.

Варіант 13. $y' = \frac{2y + y^2}{x}$, $[2,22]$ $y(2) = 8$.

Варіант 14. $y' = \frac{3y^2 - y}{2 + x}$, $[3,33]$, $y(3) = 7$.

Варіант 15. $y' = \frac{5y + y^2}{22 - x}$, $[1,21]$, $y(1) = 18$.

5.5. Контрольні запитання

1. Сформулюйте постановку задачі Коші для звичайного диференціального рівняння 1-го порядку.
2. Сформулюйте постановку задачі Коші для системи звичайних диференціальних рівнянь 1-го порядку.
3. У чому полягає ідея методу Ейлера та його модифікацій?
4. У чому полягає ідея 4-го порядку?
5. Чим відрізняються багатокрокові методи розв'язання задачі Коші від однокрокових?
6. У чому полягає ідея методу прогнозу та корекції?
7. Від чого залежить точність розв'язку задачі Коші будь-яким методом?