НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ УКРАЇНИ
«КИЇВСЬКИЙ ПОЛІТЕХНІЧНИЙ ІНСТИТУТ
ІМЕНІ ІГОРЯ СІКОРСЬКОГО»
Факультет прикладної математики
Кафедра прикладної математики

Звіт
із лабораторної роботи №3
з дисципліни «Теорія керування»
на тему
«Числове розв’язання задач варіаційного числення»

Виконала:

студентка групи КМ-01
Резниченко Є. С.

Перевірили:

Професор ПМА ФПМ
Норкін В.І.
Асистент ПМА ФПМ
Жук І.С.

Київ — 2024

Зміст

Теоретичні відомості

Розглянемо задачу варіаційного числення, тобто задачу нескінченновимірної оптимізації у просторі неперервно диференційованих функцій з фіксованими кінцями:

$$J(x(\cdot)) = \int_{\alpha}^{\beta} I(t, x(t), \dot{x}(t))dt \to \min_{x(\cdot):x(\alpha)=a, x(\beta)=b}.$$

Апроксимуємо цю задачу скінчено-вимірною задачею оптимізації наступним чином.

На інтервалі інтегрування $[\alpha, \beta]$ визначимо точки $t_i = \alpha + i(\beta - \alpha)/N$, $i = 0,..., N+1$, де $(N+1)$ - число точок, $N \sim 10$. Довжина інтервалу між сусідньома точками дорівнює $\Delta t = (\beta - \alpha)/N$.

Введемо $N-1$ змінну $x_i = x(t_i)$, $i = 1,..., N-1$, позначимо $x_0 = a$, $x_N = b$.

Апроксимуємо інтеграл в задачі варіаційного числення за формулою прямокутників:

$$J(x(\cdot)) \approx \sum_{i=0}^{N-1} I\left(t_i, x(t_i), \dot{x}(t_i)\right) \Delta t.$$

Похідну $\dot{x}(t_i)$ апроксимуємо скінченою різницею:

$$\dot{x}(t_i) \approx (x(t_{i+1}) - x(t_i))/\Delta t = (x_{i+1} - x_i)/\Delta t.$$

Друга похідна апроксимується за формулою:

$$\ddot{x}(t_i) = \left(\left(x(t_{i+1}) - x(t_i)\right)/\Delta t - \left(x(t_i) - x(t_{i-1})\right)/\Delta t\right)/\Delta t =$$
$$= \left(x(t_{i+1}) - 2x(t_i) + x(t_i)\right)/\left(\Delta t\right)^2 = \left(x_{i+1} - 2x_i + x_i\right)/\left(\Delta t\right)^2.$$

Тепер замість вихідної нескінченновимірної оптимізаційної задачі маємо оптимізаційну задачу у $(N-1)$-вимірному просторі:

$$J_N(x_1,..., x_{N-1}) = \sum_{i=0}^{N-1} I\left(t_i, x_i, (x_{i+1} - x_i)/\Delta t\right) \cdot \Delta t \to \min_{x_1,...,x_{N-1}}.$$

Також можна розглянути так звану регулярізовану задачу з додатковим квадратичним членом в цільової функції з параметром регулярізації $\lambda \geq 0$:

$$J_N^\lambda(x_1,...,x_{N-1}) = \sum_{i=0}^{N-1} I\left(t_i, x_i, (x_{i+1}-x_i)/\Delta t\right)\Delta t + \lambda \sum_{i=0}^{N-1}\left((x_{i+1}-x_i)/\Delta t\right)^2 \Delta t \to \min_{x_1,...,x_{N-1}}$$

Якщо вихідна задача містить декілька функцій $x_1(t), x_2(t),...$, то для кожної із них вводяться свої змінні, $x_{1i} = x_1(t_i)$, $x_{2i} = x_2(t_i)$. Таким чином, оптимізаційна задачу буде містити $2N$ змінних, $x_{11}, x_{12},..., x_{1N}, x_{21}, x_{22},..., x_{2N}$.

Ці задачі можна розв'язувати методами безградієнтного типу (по координатний спуск, Нелдера-Міда, випадкових напрямків, скінчено-різницевий). Альтернативно, ці задачі можна також розв'язувати методами градієнтного типу. Градієнт

$\nabla J_N(x_1,...,x_{N-1}) = \left(\dfrac{\partial J_N}{\partial x_1},...,\dfrac{\partial J_N}{\partial x_{N-1}}\right)$ функції $J_N(x_1,...,x_{N-1})$ наближено обчислюється наступним чином:

$$\frac{\partial J_N}{\partial x_i} = \left.\frac{\partial J}{\partial \dot{x}}\right|_{t=t_{i-1}}\left(\frac{1}{\Delta t}\right) + \left.\frac{\partial J}{\partial x}\right|_{t=t_i} + \left.\frac{\partial J}{\partial \dot{x}}\right|_{t=t_i}\left(\frac{-1}{\Delta t}\right) \approx$$
$$\approx \left(\frac{\partial J}{\partial \dot{x}}\left(t_{i-1}, x_{i-1}, (x_i - x_{i-1})/\Delta t\right)\right) + \left(\frac{\partial J}{\partial x}\left(t_i, x_i, (x_{i+1}-x_i)/\Delta t\right)\right)\Delta t -$$
$$-\left(\frac{\partial J}{\partial \dot{x}}\left(t_i, x_i, (x_{i+1}-x_i)/\Delta t\right)\right), \qquad i = 1,..., N-1.$$

Тут спочатку для даної підінтегральної функції $I(t, x(t), \dot{x}(t))$ аналітично обчислюються формули для часткових похідних $\dfrac{\partial J}{\partial x}$ та $\dfrac{\partial J}{\partial \dot{x}}$ як функцій змінних $(t, x(t), \dot{x}(t))$, а потім замість них підставляються трійки аргументів $\left(t_{i-1}, x_{i-1}, (x_i - x_{i-1})/\Delta t\right)$ та $\left(t_i, x_i, (x_{i+1}-x_i)/\Delta t\right)$.

Аналогічно,

$$\nabla J_N^\lambda(x_1,...,x_{N-1}) = \left(\frac{\partial J_N^\lambda}{\partial x_1},...,\frac{\partial J_N^\lambda}{\partial x_{N-1}}\right),$$

$$\frac{\partial J_N^\lambda}{\partial x_i} = \left.\frac{\partial J}{\partial \dot{x}}\right|_{t=t_{i-1}}\left(\frac{1}{\Delta t}\right) + \left.\frac{\partial J}{\partial x}\right|_{t=t_i} + \left.\frac{\partial J}{\partial \dot{x}}\right|_{t=t_i}\left(\frac{-1}{\Delta t}\right) + \frac{2\lambda}{(\Delta t)}\left(x_i - x_{i-1}\right) - \frac{2\lambda}{(\Delta t)}\left(x_{i+1} - x_i\right) \approx$$
$$\approx \left(\frac{\partial J}{\partial \dot{x}}\left(t_{i-1}, x_{i-1}, (x_i - x_{i-1})/\Delta t\right)\right) + \left(\frac{\partial J}{\partial x}\left(t_i, x_i, (x_{i+1}-x_i)/\Delta t\right)\right)\Delta t -$$
$$-\left(\frac{\partial J}{\partial \dot{x}}\left(t_i, x_i, (x_{i+1}-x_i)/\Delta t\right)\right) + \frac{2\lambda}{(\Delta t)}\left(x_i - x_{i-1}\right) - \frac{2\lambda}{(\Delta t)}\left(x_{i+1} - x_i\right).$$

Порядок виконання роботи

1) Обрати приклад задачі варіаційного числення зі списку та повідомить викладача.
2) Обрати число точок апроксимації N.
3) Запрограмувати функцію $J_N(x_1,...,x_{N-1})$.
4) Обрати метод безградієнтний (або градієнтний) метод мінімізації $J_N(x_1,...,x_{N-1})$.
5) Застосувати метод до мінімізації $J_N(x_1,...,x_{N-1})$ та побудувати графік залежності отриманого значення мінімізованої функції від числа ітерацій, обрати критерій зупинки методу.
6) Для отриманого розв'язку $(x_1^*,...,x_{N-1}^*)$ (мінімуму) побудувати кусково-лінійну функцію по точкам $\left((\alpha,a),(t_1,x_1^*),...,(t_{N-1},x_{N-1}^*),(\beta,b)\right)$ та зобразити її графічно.
7) Графічно порівняти отриманий наближений розв'язок з точним розв'язком, отриманим аналітично.
8) Дослідити точність апроксимації від числа точок апроксимації.
9) Підготувати письмовий звіт про роботу (та надіслати викладачу).

Відповідно до варіанту №15 було обрано наступну задачу:

15. $\int_0^1((x')^2+4x^2)dt \to extr$, $x(0)=e^2$, $x(1)=1$.

Основна частина

У даній лабораторній роботі було зафіксовано наступні константи:

$$N = 15$$

$$A = 0$$

$$B = 1$$

$$X(0) = e^2$$

$$X(1) = 1$$

$$FUNC = "x' ** 2 + 4 * x ** 2"$$

Після цього було запрограмовано функцію $J_N(x_1, ..., x_{N-1})$ відповідно до теоретичних відомостей.

Було використано функцію optimize.minimize із бібліотеки Scipy, яка використовуючи градієнтний метод оптимізує задану функцію.

Щоб порівняти результати, спочатку потрібно вручну розв'язати задачу та отримати аналітичний розв’язок. Після цього можна провести графічне порівняння отриманих результатів.

За формулою Ейлера:

$$F_{x'x'} * x'' + F_{xx'} * x' + F_{tx'} - F_x = 0$$

$$F_x(t, x, x') = 8x$$

$$F_{xx'}(t, x, x') = 0$$

$$F_{x'}(t, x, x') = 2x'$$

$$F_{x'x'}(t, x, x') = 2$$

$$F_t(t, x, x') = 0$$

$$F_{tx'}(t, x, x') = 0$$

Отримаємо рівняння:

$$2x'' - 8x = 0$$

Характеристичне рівняння:

$$2\lambda^2 - 8 = 0$$

$$\lambda_{1,2} = \pm\sqrt{4}$$

$$y = C_1 e^{-2x} + C_2 e^{2x}$$

$$\begin{cases} x(0) = e^2 \\ x(1) = 1 \end{cases}$$

$$\begin{cases} C_1 e^0 + C_2 e^0 = e^2 \\ C_1 e^{-2} + C_2 e^2 = 1 \end{cases}$$

$$C_1 = e^2, \quad C_2 = 0$$

$$y = e^2 e^{-2x}$$

Отримавши точний розв'язок задачі, наступним кроком буде його порівняння з наближеним результатом.

У ході досліджень наближеного розв’язку було отримано такі результати:

Значення мінімізованої функції: 7.710237

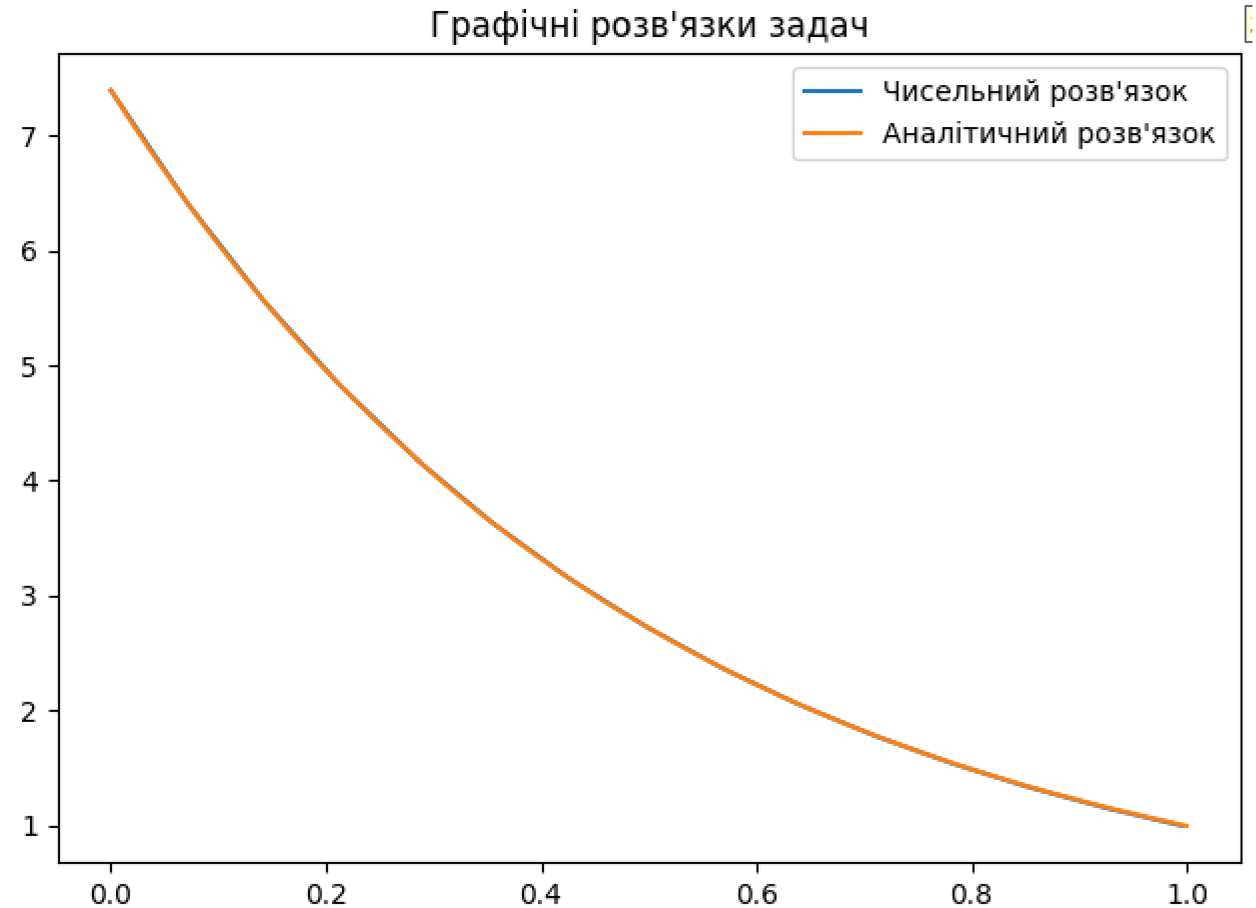

Рис. 1 – Чисельний та аналітичний розв’язок задачі при N=15

Точність такого розв’язку велика, але дослідимо, як зміниться значення мінімізованої функції, якщо вдвічі збільшити значення параметру N та яким буде графік.

Збільшимо параметр N до 30:

Значення мінімізованої функції: 3.707571

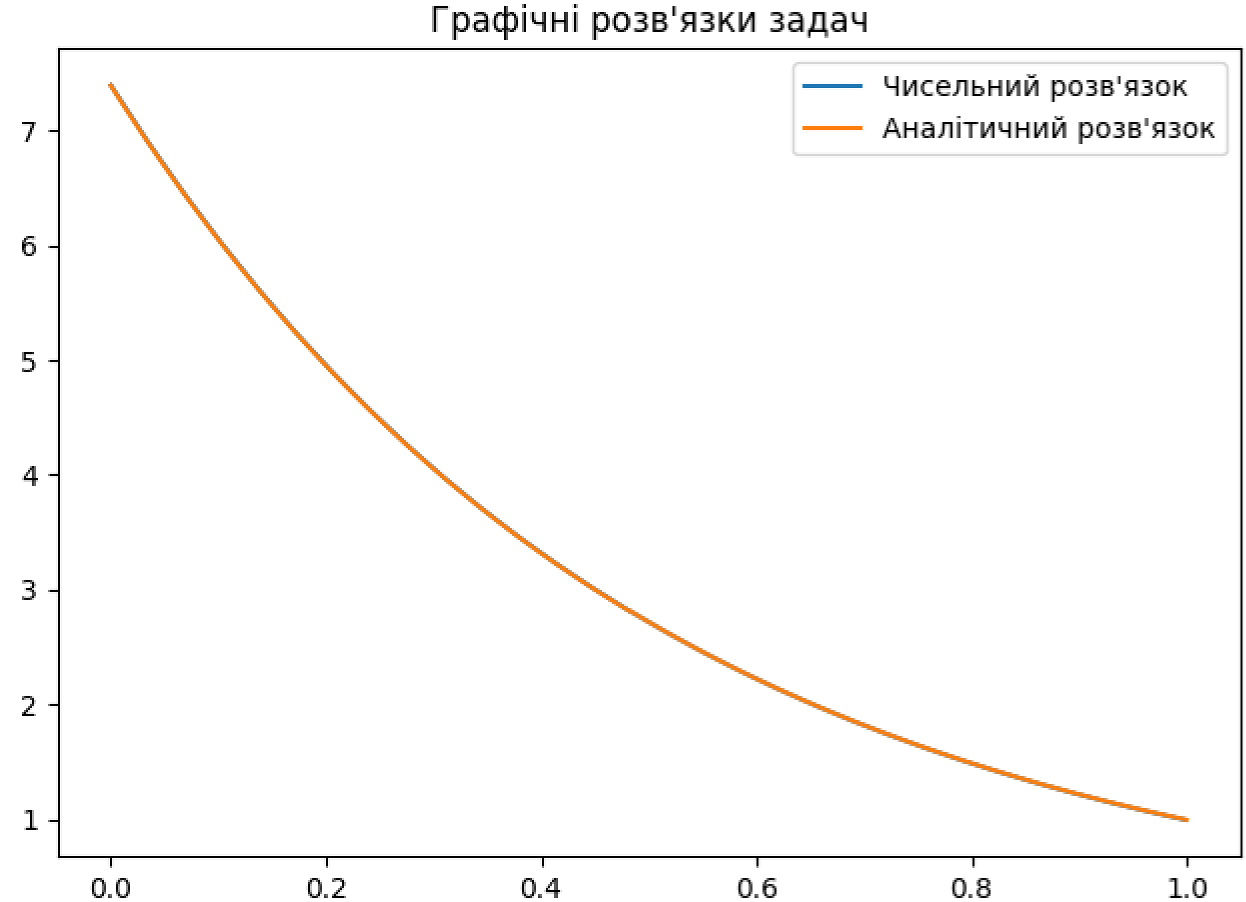

Рис. 2 – Чисельний та аналітичний розв’язок задачі при N=30

Таким чином, при N=30 або більше, відмінності між наближеним та точним розв'язками стають ще більш незначними.

Висновок

У процесі виконання цієї лабораторної роботи задачу варіаційного числення було розв'язано аналітично та чисельно.

Були побудовані графіки розв'язків. Висновок полягає в тому, що при N=30 наближений розв'язок значно збігається з точним розв'язком задачі. За потреби можна зменшити або збільшити параметр N за необхідності.

Використана література

1. Методичні вказівки до лабораторної роботи.
2. Лейтман Дж. Введение в теорию оптимального управления. М.: Наука, 1968. 192 стр.

Додаток А – Код програми

```
import numpy as np
import scipy.optimize
import matplotlib.pyplot as plt


PRECISION = 6
np.set_printoptions(precision=PRECISION)


def j(args):
    global x_list
    x_list = []
    _b = B + step
    dt = (_b-A)/N

    func_str = ""
    for i in range(1, N + 2):
        x_list.append(A + (i - 1) * dt)
        func_str += "(" + FUNC.replace("x'", f"((y{i + 1}-y{i})/{dt})")\
                              .replace("x", f"(x{i})")\
                              .replace("t", f"{dt}") + f")*{dt}"
        if i != N+1:
            func_str += "+"
    func_str = func_str.replace("y", "x")
    func_str = f"(1/{N})*(" + func_str + ")"

    f_str = func_str

    f_str = f_str.replace(f'x{N}', str(YB))
    for i in range(len(args), 1, -1):
        f_str = f_str.replace(f"x{i}", str(args[i - 1]))
    f_str = f_str.replace('x1', str(YA))
    return eval(f_str)


def variant_15(n):
    x0 = np.array([1 for _ in range(n + 2)])
    res = scipy.optimize.minimize(j, x0)
    print(f"Значення мінімізованої функції: {round(res.fun, PRECISION)}")

    result = res.x[1:-2]
    result[-1] = YB
    y_list = np.concatenate([[YA], result])
```

```python
    t_plot = np.linspace(A, B, 200)
    analytic = [np.exp(2*(1-t)) for t in t_plot]

    plt.plot(x_list[:-1], y_list, label="Чисельний розв'язок")
    plt.plot(t_plot, analytic, label="Аналітичний розв'язок")
    plt.title("Графічні розв'язки задач")
    plt.legend()
    plt.tight_layout()
    plt.show()


if __name__ == "__main__":
    N = 30              #  Число точок апроксимації
    A = 0               #  Нижня межа інтегрування
    B = 1               #  Верхня межа інтегрування
    YA = np.exp(2)      #  x(0) = e^2
    YB = 1              #  x(1) = 1
    FUNC = "x'**2+4*x**2"

    step = (B-A)/N
    variant_15(N)
```