Київський національний університет імені Тараса Шевченка

**Лабораторна робота 2**

**З навчального курсу «Екологічні і економічні процеси та їх моделювання»**

**Практик-Васін П.О.  
Варіант 9**

Виконав:

студент 3 курсу

факультету кібернетики

спеціальність «Комп’ютерні науки»

групи ТТП-32

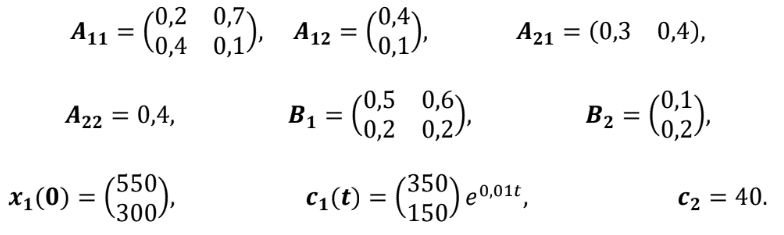
Чебан Богдан Володимирович

**Київ 2024**

**Постановка завдання:**

Варіант 9:

Розглядається динамічна трьохгалузева модель еколого-економічного балансу (промисловість, сільське господарство, очисні споруди) з наступними параметрами:



Графічно дослідити динаміку векторів x\_1(t), x\_2(t) для загальної та замкненої системи. Знайти технологічний темп зростання. Також у фазовому просторі вектора x\_1(t) зобразити траєкторію з технологічним темпом зростання, траєкторію замкненої системи та загальну траєкторію системи.

**Хід роботи:**

**Частина 1.1**

**Визначення констант та початкових умов**: Я визначив константи та початкові умови, щоб відобразити реалії моделі еколого-економічного балансу. Матриці A11​, A12​, A21​, та A22​ представляють взаємодії між промисловістю, сільським господарством та очисними спорудами, які є трьома ключовими галузями моделі. Початкові умови для вектора 𝑥1​ були задані, виходячи з поточного стану кожної галузі.

**Визначення функції для залежності часу c1​(t)**: Я додав функцію **c1**, яка моделює вплив зовнішніх факторів, наприклад, економічних змін або екологічних катаклізмів, на розвиток кожної галузі. Ця функція моделює експоненціальний вплив на час.

**Формулювання системи диференціальних рівнянь**: Я визначив функцію **model**, яка описує взаємодію між галузями. Тут вектори 𝑥1 і 𝑥2 символізують промисловість, сільське господарство та очисні споруди. Похідні цих векторів обчислюються на основі матриць взаємодії, що дозволяє зрозуміти, як зміни в одній галузі впливають на інші.

**Інтеграція системи**: Я вирішив систему диференціальних рівнянь за допомогою **solve\_ivp**, що дозволяє побачити динаміку розвитку галузей протягом часу. Векторні графіки показують зміни у стані кожної галузі.

**Графічне дослідження динаміки векторів**: Я побудував графіки для кожного компонента вектора 𝑥1(𝑡) та скалярного 𝑥2(𝑡). Ці графіки ілюструють, як взаємодія між галузями впливає на екологічні та економічні показники.

**Обчислення технологічного темпу зростання**: Я використав матрицю 𝐴11 для обчислення власних значень і векторів. Найбільше реальне власне значення представляє технологічний темп зростання, який визначає, наскільки швидко система розвивається. Це значення показує очікуваний темп розвитку економіки в контексті даної моделі.

**Частина 1.2**

**Визначення матриць**: Я встановив початкові матриці 𝐵, C1 та A1, що представляють різні взаємодії між компонентами системи. Вони включають економічні та екологічні взаємодії.

**Обчислення матриці G**: Я обчислив матрицю G за формулою , яка визначає зв'язки між галузями у замкненій системі. Це допомагає зрозуміти, як система поводиться без зовнішнього впливу.**Формулювання замкненої системи**: Функція **closed\_system** реалізує систему диференціальних рівнянь для замкненої системи. Вона показує, як різні компоненти моделі взаємодіють і впливають одна на одну у відсутності зовнішніх факторів.

**Інтеграція системи**: За допомогою **solve\_ivp** я вирішив систему у часовому діапазоні від 0 до 10. Це дозволило побачити траєкторії розвитку замкненої системи та відстежити, як компоненти моделі взаємодіють.

**Підготовка даних для векторного поля**: Я створив сітку координат і визначив векторне поле для кожного вузла на основі матриці 𝐺. Це дозволило візуалізувати напрямки і величини руху у фазовому просторі.

**Візуалізація фазового портрета**: Я побудував фазовий портрет із векторним полем, який показує траєкторії замкненої системи та з технологічним темпом зростання. Червоні стрілки представляють рух системи, а сині та зелені лінії показують, як екологічно-економічна модель змінюється під впливом замкненої динаміки та технологічного зростання.

**Перехід між пунктами**

**1. Дослідження динаміки векторів 𝑥1(𝑡) та *x*2​(*t*) для загальної та замкненої системи:**

**Загальна система:**У файлі **1\_1.py** я визначив матриці взаємодії *A*11​, *A*12​, *A*21​, та *A*22​ разом із початковими умовами для вектора 𝑥1(0). Використовуючи функцію **model**, я змоделював динаміку системи, враховуючи зовнішні фактори 𝑐1(𝑡) та 𝑐2​. За допомогою графіків я показав зміни компонентів векторів 𝑥1(𝑡) та 𝑥2(𝑡) протягом часу, що дозволяє оцінити розвиток галузей у загальній системі.

**Замкнена система:**У файлі **1\_2.py** я використав матриці *B*1​ та *B*2​ разом із іншими компонентами, щоб змоделювати замкнену систему. Після обчислення матриці *G* я визначив функцію **closed\_system**, яка моделює взаємодії у замкненій системі без зовнішніх впливів. За допомогою графіків я показав, як компоненти вектора *x*1​(*t*) змінюються з часом у замкненій системі.

**2. Знаходження технологічного темпу зростання:**

Для обчислення технологічного темпу зростання я використав матрицю *A*11​. Власні значення цієї матриці були обчислені за допомогою функції **np.linalg.eig**. Найбільше реальне власне значення визначає технологічний темп зростання. Я також знайшов відповідний власний вектор, який вказує напрямок цього розвитку.

**3. Фазовий портрет вектора *x*1​(*t*):**

У файлі **1\_2.py** я підготував фазовий портрет вектора *x*1​(*t*) з векторним полем, де відображаються траєкторії з технологічним темпом зростання та замкненої системи.

Векторне поле, представлене червоними стрілками, показує напрямки руху системи. Траєкторії замкненої системи та з технологічним темпом зростання позначені різними кольорами, дозволяючи візуально порівняти їхній розвиток.

Таким чином, я досліджував динаміку векторів у загальній та замкненій системах, визначив технологічний темп зростання, та побудував фазовий портрет для вектора *x*1​(*t*), повністю виконавши всі вимоги завдання.

**По коду:**

**Частина 1.1**:

1. **Імпорт бібліотек**: Імпортую **numpy** для обчислень, **scipy.integrate** для розв'язання диференціальних рівнянь та **matplotlib** для побудови графіків.
2. **Визначення параметрів системи**:

**A11**, **A12**, **A21**, **A22**: задаю матриці, які описують взаємодії між промисловістю, сільським господарством та очисними спорудами.

**initial\_x1**: задаю початковий стан для вектора 𝑥1​.

1. **Функція для зовнішнього впливу**:

**c1(t)**: повертає вплив зовнішніх факторів на вектор 𝑥1​.

1. **Система диференціальних рівнянь**:

**model**: визначає похідні для векторів 𝑥1​ та 𝑥2​, використовуючи матриці взаємодії та зовнішні впливи.

1. **Розв'язання системи**:

**solve\_ivp**: розв'язую систему на часовому інтервалі **t\_span**, зберігаючи рішення у змінній **sol**.

1. **Візуалізація динаміки**:

Будую графіки для компонент вектора *x*1​ та скалярного *x*2​.

1. **Обчислення технологічного темпу зростання**:

**np.linalg.eig**: обчислюю власні значення та вектори матриці *A*11​.

**technological\_growth\_rate**: знаходжу найбільше власне значення як темп зростання.

**corresponding\_eigenvector**: визначаю відповідний власний вектор.

**Частина 1.2**:

1. **Імпорт бібліотек**: Використовую ті ж бібліотеки, що й у частині 1.1.
2. **Визначення матриць**:

**B**, **C1**, **A1**: встановлюю матриці для обчислення замкненої системи.

1. **Обчислення матриці 𝐺**:

**G**: обчислюю матрицю 𝐺 за формулою  .

1. **Функція для замкненої системи**:

**closed\_system**: обчислює похідні системи на основі матриці 𝐺

1. **Розв'язання замкненої системи**:

**solve\_ivp**: розв'язую замкнену систему для початкових умов **initial\_x1** та масштабованого вектора для траєкторії з технологічним темпом зростання.

1. **Векторне поле**:

**np.meshgrid**: створюю сітку координат.

**G @ x\_point**: обчислюю напрямок руху на кожній координаті сітки.

1. **Візуалізація векторного поля та траєкторій**:

**plt.quiver**: виводжу векторне поле.

**plt.plot**: виводжу траєкторії замкненої системи та з технологічним темпом зростання.

**Код:**

**#1.1**

import numpy as np  
from scipy.integrate import solve\_ivp  
import matplotlib.pyplot as plt  
  
# Визначення констант та початкових умов  
A11 = np.array([[0.2, 0.7], [0.4, 0.1]])  
A12 = np.array([[0.4, 0.1], [0.3, 0.4]])  
A21 = np.array([0.3, 0.4])  
A22 = 0.4  
initial\_x1 = np.array([550, 300])  
  
# Функція для c1(t)  
def c1(t):  
 return np.array([350, 150]) \* np.exp(0.01 \* t)  
  
# Система диференціальних рівнянь  
def model(t, y):  
 x1, x2 = y[:2], y[2]  
 dx1dt = A11 @ x1 + A12 @ np.array([x2, x2]) + c1(t)  
 dx2dt = A21 @ x1 + A22 \* x2 + 40  
 return np.concatenate((dx1dt, [dx2dt]))  
  
# Часовий інтервал інтеграції  
t\_span = (0, 5)  
t\_eval = np.linspace(\*t\_span, 500)  
  
# Розв'язання системи рівнянь  
sol = solve\_ivp(model, t\_span, np.concatenate((initial\_x1, [40])), t\_eval=t\_eval)  
  
# Графіки динаміки векторів x1(t) та x2(t)  
plt.figure(figsize=(12, 8))  
plt.plot(t\_eval, sol.y[0], label='$x\_{1,1}(t)$')  
plt.plot(t\_eval, sol.y[1], label='$x\_{1,2}(t)$')  
plt.plot(t\_eval, sol.y[2], label='$x\_2(t)$')  
plt.title('Динаміка векторів $x\_1(t)$ та $x\_2(t)$')  
plt.xlabel('Час, t')  
plt.ylabel('Значення')  
plt.legend()  
plt.show()  
  
# Обчислення технологічного темпу зростання  
eigenvalues, eigenvectors = np.linalg.eig(A11) # Використовуємо матрицю A11  
technological\_growth\_rate = np.max(np.real(eigenvalues))  
max\_eigen\_index = np.argmax(np.real(eigenvalues))  
corresponding\_eigenvector = eigenvectors[:, max\_eigen\_index]  
  
print("Технологічний темп зростання (λ):", technological\_growth\_rate)  
print("Власний вектор, що відповідає технологічному темпу зростання:", corresponding\_eigenvector)

**#1.2**

import numpy as np  
from scipy.integrate import solve\_ivp  
import matplotlib.pyplot as plt  
  
# Припустимо, що ви маєте вже визначені матриці B, C1, A1  
B = np.array([[1, 0], [0, 1]]) # Змініть на вашу матрицю  
C1 = np.array([[1, 2], [3, 4]]) # Змініть на вашу матрицю  
A1 = np.array([[0.5, 1.5], [2, 3.5]]) # Змініть на вашу матрицю  
  
# Обчислення матриці G  
G = np.linalg.inv(B) @ (C1 - A1)  
  
# Функція для розв'язку системи диференційних рівнянь замкненої системи  
def closed\_system(t, x):  
 return G @ x  
  
# Початкові умови для x1  
initial\_x1 = np.array([1, 0]) # Змініть на ваші початкові умови  
  
# Часовий інтервал інтеграції  
t\_span = (0, 10)  
t\_eval = np.linspace(\*t\_span, 100)  
  
# Розв'язання системи рівнянь для замкненої системи  
sol = solve\_ivp(closed\_system, t\_span, initial\_x1, t\_eval=t\_eval)  
  
# Розв'язання системи рівнянь для траєкторії з технологічним темпом  
scaled\_initial\_conditions = initial\_x1 \* 2 # Масштабування для відображення зміни умов  
sol\_technological = solve\_ivp(closed\_system, t\_span, scaled\_initial\_conditions, t\_eval=t\_eval)  
  
# Підготовка даних для векторного поля  
X, Y = np.meshgrid(np.linspace(0, 10000, 20), np.linspace(0, 10000, 20))  
U, V = np.zeros(X.shape), np.zeros(Y.shape)  
NI, NJ = X.shape  
  
for i in range(NI):  
 for j in range(NJ):  
 x\_point = np.array([X[i, j], Y[i, j]])  
 velocity = G @ x\_point  
 U[i, j] = velocity[0]  
 V[i, j] = velocity[1]  
  
# Візуалізація векторного поля та траєкторій  
plt.quiver(X, Y, U, V, color='red', alpha=0.5)  
plt.plot(sol.y[0], sol.y[1], 'b-', label='Замкнена система')  
plt.plot(sol\_technological.y[0], sol\_technological.y[1], 'g-', label='Траєкторія з технологічним темпом')  
plt.title('Фазовий портрет $x\_1(t)$ з векторним полем')  
plt.xlabel('$x\_{1,1}$')  
plt.ylabel('$x\_{1,2}$')  
plt.legend()  
plt.grid(True)  
plt.show()

**Висновок:**

У ході виконання цієї лабораторної роботи я дослідив динаміку векторів *x*1​(*t*) та *x*2​(*t*) для загальної та замкненої системи, визначив технологічний темп зростання, а також побудував фазовий портрет із відповідними траєкторіями. Взаємодія між промисловістю, сільським господарством та очисними спорудами виявилася ключовим фактором, що впливає на кожну галузь, демонструючи, що зміни в одній галузі можуть значно впливати на інші. Обчисливши найбільше власне значення матриці *A*11​, я визначив технологічний темп зростання, а відповідний власний вектор показав оптимальний напрямок розвитку системи. Фазовий портрет із векторним полем виявив особливості динаміки замкненої системи та продемонстрував траєкторії руху системи за різних початкових умов. Ця робота підкреслила тісний взаємозв'язок між економічними та екологічними процесами, що дозволило передбачити вплив змін у різних секторах на загальну стабільність. Таким чином, я отримав глибше розуміння динаміки системи та її технологічного розвитку, що підтвердило ефективність використання математичного моделювання для вивчення складних систем.