

РОЗРОБКА ПРОГРАМНОГО WEB-ІНСТРУМЕНТУ ДЛЯ СПЕКТРАЛЬНОГО АНАЛІЗУ ТА ПОРІВНЯННЯ ГРАФІВ

Леонтюк Н. С., Соломко В. О.

Київський столичний університет імені Бориса Грінченка, м. Київ

ВСТУП

Графи є універсальним математичним інструментом для моделювання зв'язків між об'єктами у найрізноманітніших системах. Вони описують структури соціальних мереж, транспортних шляхів, інформаційних потоків, біологічних взаємодій, схем зв'язку та безліч інших процесів. Будь-яка складна система, де існують елементи та відношення між ними, може бути інтерпретована як граф (наприклад, [1]).

Актуальність і постановка проблеми. У спектральній теорії графів центральне місце займає дослідження властивостей матриць, які описують структуру графа, зокрема їх власних значень та власних векторів. Власні значення різних матриць, пов'язаних із графом, кодують важливі структурні характеристики: кількість компонент зв'язності, центральність вершин, наявність кластерів, симетрії тощо (наприклад, [2]). Саме на цих властивостях базуються методи спектральної кластеризації, що широко використовуються у сучасному аналізі даних.

Одним із цікавих аспектів спектральної теорії є явище коспектральності, коли різні (неізоморфні) графи мають однаковий спектр. Вперше такі приклади були відомі ще з середини ХХ століття, а їхнє системне дослідження узагальнене у монографії “Spectra of Graphs”[3]. Коспектральність відіграє ключову роль у розумінні того, які властивості графів можуть бути повністю «зчитані» зі спектра, а які - ні.

Мета дослідження. Мета роботи полягає в адаптації та алгоритмічній реалізації існуючих методів спектрального аналізу графів, у тому числі визначення коспектральності графів, а також у проведенні експериментального аналізу їх роботи на практичних прикладах за допомогою створеного веб-інструменту.

Аналіз останніх досліджень і публікацій. Перші дослідження коспектральності пов'язані з теорією графів у хімії протягом 1950–1960-х років. У роботах Гюнхарда і Примаса було доведено, що неізоморфні молекулярні структури можуть мати однакові спектральні характеристики. Це означало, що спектр не може бути “унікальним відбитком” структури графа, і саме це призвело до формування поняття коспектральних графів. Перший приклад неізоморфних коспектральних дерев навели Колатц і Сіноговіц у своїй праці “Spektren endlicher Graphen”[4]. Важливий внесок у алгоритмічний аналіз графів зробив Брендан Мак-Кей, який у роботі “Computer reconstruction of small graphs” [5] запропонував ефективні методи генерування та перевірки малих неізоморфних графів. Запропоновані ним алгоритми стали основою для сучасних інструментів

дослідження графових інваріантів, включно зі спектральними. Ця праця є фундаментальною для розуміння меж графових інваріантів, зокрема ситуацій, коли різні графи не можуть бути розрізнені стандартними методами, що безпосередньо пов'язано з явищем коспектральності.

Сучасні дослідження у галузі спектральної теорії графів значною мірою зосереджені на вивченні коспектральності, побудові коспектральних пар графів та аналізі структурних властивостей, які не можуть бути повністю визначені спектром графа. У роботі [6] запропоновано підхід до побудови коспектральних графів відносно узагальненої матриці суміжності. Автори розширюють класичне поняття суміжності, вводячи вагові та структурні модифікації матриці, що дозволяє отримувати нові сім'ї коспектральних графів. Цей результат є важливим для розуміння того, як спектральні методи можуть застосовуватись у контекстах, де структура графа виходить за межі простої бінарної моделі. У дослідженні [2] наведено приклад побудови пари коспектральних 5-регулярних графів із різними структурними властивостями, зокрема один граф має досконале парування, тоді як інший — ні. Це підкреслює, що спектр не є повним інваріантом графа, навіть у класі регулярних структур, і демонструє межі спектральних методів при аналізі топологічних характеристик. Публікація [1] розширює спектральний аналіз на графи, пов'язані з алгебраїчними структурами, зокрема унітарні графи Келі кілець верхньотрикутних матриць. Ця робота демонструє загальність спектрального підходу та можливість його застосування до нетривіальних математичних моделей, що робить спектральні методи перспективними і для задач кластеризації структурованих даних. Загалом, аналіз сучасних публікацій свідчить про активний розвиток досліджень у напрямку коспектральності графів та алгоритмів спектрального аналізу. Отримані в останніх роботах результати підтверджують важливість спектра як інструменту, але одночасно наголошують на необхідності обережної інтерпретації його можливостей.

РЕЗУЛЬТАТИ ДОСЛІДЖЕННЯ

Нехай Γ – неорієнтований граф без кратних ребер і петель. Наведемо означення основних матриць, асоційованих з графом.

1. Матриця суміжності

Матриця суміжності $A = (a_{ij})_{i,j=1}^n$ – це матриця розміру $n \times n$, де n — кількість вершин графа Γ , а елементи визначаються як

$$a_{ij} = \begin{cases} 1, & \text{якщо вершини } i \text{ та } j \text{ з'єднані ребром,} \\ 0, & \text{інакше.} \end{cases}$$

2. Матриця Лапласа

Для неорієнтованого графа без петель матриця Лапласа (Лапласіан) визначається як:

$$L = D - A,$$

де D - діагональна матриця, у якої діагональні елементи – степені відповідних вершин графа Γ .

3. Матриця Зейделя

Матриці Зейделя $S = (s_{ij})_{i,j=1}^n$ - це матриця розміру $n \times n$, де n — кількість вершин графа Γ , а елементи визначаються як

$$s_{ij} = \begin{cases} 0, & \text{якщо } i = j, \\ -1, & \text{якщо вершини } i \text{ та } j \text{ з'єднані ребром,} \\ 1, & \text{якщо вершини } i \text{ та } j \text{ не з'єднані.} \end{cases}$$

Для матриці M характеристичний многочлен визначається як:

$$p_M(\theta) = \det(\theta I - M).$$

Корені цього многочлена є власними значеннями матриці M , де $M = A, M = L$ або $M = S$ відповідно. Множину власних значень матриці M з урахуванням їх кратностей називають спектром графа Γ відносно матриці M та позначають $\sigma_\Gamma(M)$.

Два неізоморфні графи називають коспектральними, якщо вони мають однаковий спектр. Побудова коспектральних графів відносно будь-якої узагальненої матриці суміжності описана у [6].

Коспектральність показує межі того, що можна визначити через спектр:

- спектр визначає деякі властивості (наприклад, регулярність, кількість вершин, суму степенів),
- але не визначає структуру однозначно.

У межах дослідження була реалізована програмна система для автоматичної перевірки коспектральності графів за їхніми матрицями. Інструмент працює у форматі Web-додатку, створеного на основі Python-фреймворку Streamlit, що забезпечує інтерактивну взаємодію з користувачем без необхідності встановлення додаткових бібліотек або IDE.

Створений Web-додаток реалізує такі можливості:

1. Завантаження двох матриць графів у форматі CSV
 - матриця суміжності;
 - матриця Лапласа;
 - матриця Зейделя.
 2. Автоматична перевірка коректності даних
 - квадратність матриць;
 - однаковий розмір обох матриць;
 - числовий формат даних.
 3. Обчислення спектрів обох графів
 - знаходження власних значень;
 - порівняння множин власних значень із точністю до допустимої похибки.
 4. Виведення результату
 5. Візуальне представлення вихідних матриць
- Проект складається з двох основних файлів:

app.py - Web-інтерфейс (Рис.1)

Відповідає за:

- завантаження CSV-файлів;
- відображення матриць;
- виклик функції спектрального аналізу;
- показ результатів.

```

import streamlit as st
import pandas as pd
import numpy as np

from cospectral import are_cospectral

st.title("Перевірка матриць на коспектральність")

uploaded_A = st.file_uploader("Завантажте матрицю А (CSV)", type=["csv"])
uploaded_B = st.file_uploader("Завантажте матрицю В (CSV)", type=["csv"])

def load_matrix(file):
    """Завантаження CSV-файла із автовизначенням розподільовача та
    очисткою"""
    try:
        df = pd.read_csv(file, header=None)
    except:
        df = pd.read_csv(file, header=None, sep=';')

    # Видалення пустих рядків
    df = df.dropna(axis=0, how='all').dropna(axis=1, how='all')

    # Перетворення у формат числа, в разі знаходження пропусків
    df = df.apply(pd.to_numeric, errors='coerce')

    return df

if uploaded_A and uploaded_B:

```

Рис. 1. Код app.py

cospectral.py - математичний модуль (Рис.2)

Містить логіку:

- перевірки коректності матриць;
- обчислення власних значень;
- порівняння спектрів.

```

import numpy as np

def are_cospectral(A, B, tol=1e-8):
    if A.shape != B.shape:
        return False

    # Перевірка матриці на квадратність
    if A.shape[0] != A.shape[1]:
        raise ValueError("Матриця A не є квадратною.")
    if B.shape[0] != B.shape[1]:
        raise ValueError("Матриця B не є квадратною.")

    # Власні значення
    eigA = np.linalg.eigvals(A)
    eigB = np.linalg.eigvals(B)

    # Сортування
    eigA = np.sort(np.real_if_close(eigA))
    eigB = np.sort(np.real_if_close(eigB))

    # Порівняння
    return np.allclose(eigA, eigB, atol=tol)

```

Рис. 2. Код cospectral.py

CSV-файли містять матриці у стандартному вигляді, розділені комами, наприклад:

```

0,1,0
1,0,1
0,1,0

```

Такий формат забезпечує універсальність, оскільки може бути створений:

- у Excel,
- у Google Sheets,
- у будь-якому текстовому редакторі.

Алгоритмічна реалізація перевірки коспектральності:

Приклад 1. Розглянемо два неізоморфні графи на 5 вершинах:

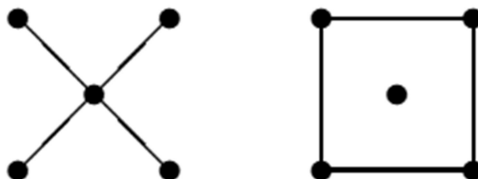


Рис. 3. Два неізоморфні графи на 5 вершинах

Їх матриці суміжності А і В мають вигляд:

$$A = \begin{bmatrix} 0 & 1 & 0 & 1 & 0 \\ 1 & 0 & 1 & 0 & 0 \\ 0 & 1 & 0 & 1 & 0 \\ 1 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \end{bmatrix} \quad B = \begin{bmatrix} 0 & 1 & 1 & 1 & 1 \\ 1 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 \end{bmatrix}$$

Крок 1. Завантаження та перевірка матриць:

- Перевірка квадратності;
- Перевірка розміру;
- Перевірка наявності лише числових значень.

Завантажте матрицю А (CSV)

Drag and drop file here
Limit 200MB per file • CSV

Browse files

A2.csv 55.0B

Завантажте матрицю В (CSV)

Drag and drop file here
Limit 200MB per file • CSV

Browse files

B2.csv 55.0B

Форма матриці А: (5, 5)

Форма матриці В: (5, 5)

Матриця А:

	0	1	2	3	4
0	0	1	0	1	0
1	1	0	1	0	0
2	0	1	0	1	0
3	1	0	1	0	0
4	0	0	0	0	0

Матриця В:

	0	1	2	3	4
0	0	1	1	1	1
1	1	0	0	0	0
2	1	0	0	0	0
3	1	0	0	0	0
4	1	0	0	0	0

Рис. 4. Вигляд завантажених матриць А та В в інструменті

Крок 2. Обчислення спектра:

$$\sigma(A) = \{2, 0, 0, 0, -2\}$$

$$\sigma(B) = \{2, 0, 0, 0, -2\}$$

Крок 3. Сортування власних значень:

Необхідно для коректного порівняння.

$$\sigma(A) = [-2, 0, 0, 0, 2]$$

$$\sigma(B) = [-2, 0, 0, 0, 2]$$

Крок 4. Порівняння спектрів із похибкою:

$$\sigma(A) = \sigma(B)$$

Крок 5. Повернення булевого результату:

Оскільки власні значення збігаються, то алгоритм повертає True:

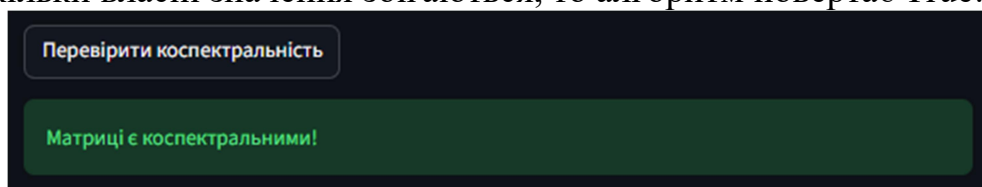


Рис. 5. Результат порівняння спектрів матриць A та B

ВИСНОВКИ

Дослідження спектра графів дозволяє аналізувати структуру мереж, виявляти взаємозв'язки між вершинами, визначати кластери та виявляти приховані симетрії.

У контексті даної наукової роботи особливу увагу приділено:

- явищу коспектральності графів,
- практичній реалізації алгоритмів спектрального аналізу.

Практична частина роботи полягала у створенні програмного інструменту для спектрального аналізу графів. На основі фреймворку Streamlit реалізовано веб-додаток, який забезпечує:

- завантаження матриць графів у форматі CSV;
- автоматичну перевірку квадратності та коректності даних;
- обчислення власних значень матриць;
- визначення коспектральності графів шляхом порівняння їхніх спектрів;
- візуалізацію вихідних матриць та інтерактивний інтерфейс користувача.

Інструмент був протестований на класичних прикладах коспектральних графів з літератури, що підтвердило коректність реалізованого алгоритму. Наявність переносної ("portable") версії забезпечує можливість запуску програми без встановлення інтерпретатора Python та додаткових бібліотек, що робить систему зручною для демонстрацій та використання іншими дослідниками.

Другий етап дослідження передбачає розширення функціоналу веб-додатку до повноцінного інструменту для спектральної кластеризації даних, включно з побудовою графів за матрицями подібності, обчисленням Лапласових власних векторів та автоматичним розподілом об'єктів на кластери. Таким чином, робота поєднує глибоке теоретичне опрацювання

спектральних методів та їх практичну реалізацію в сучасному інструментарії аналізу даних.

ДЖЕРЕЛА

1. Hołubowski W., Oliynyk B., Solomko V. On the Characterization of the Unitary Cayley Graphs of the Upper Triangular Matrix Rings. *Symmetry*. 2025. 17(12), 2180. URL: <https://doi.org/10.3390/sym17122180>
2. Соколов В., Соломко В. Побудова пари коспектральних 5-регулярних графів, один з яких має досконале паруння, а інший – ні. *Могиланський математичний журнал*. 2021. Том 4. С. 24 – 27. URL: <https://doi.org/10.18523/2617-70804202124-27>
3. Brouwer A.E., Haemers W.H. *Spectra of Graphs*. Springer: New York, NY, USA, 2012. URL: <https://doi.org/10.1007/978-1-4614-1939-6>
4. Von Collatz L., Sinogowitz U. Spektren endlicher grafen. *Abhandlungen aus dem Mathematischen Seminar der Universität Hamburg*. 1957. Vol. 21, no. 1. P. 63–77. URL: <https://doi.org/10.1007/bf02941924>
5. McKay B. D. Computer reconstruction of small graphs. *Journal of Graph Theory*. 1977. Vol. 1, no. 3. P. 281–283. URL: <https://doi.org/10.1002/jgt.3190010309>
6. Грушка Д., Лебідь В. Побудова коспектральних графів відносно узагальненої матриці суміжності. *Могиланський математичний журнал*. 2018. Том 1. С. 11 – 14. URL: <https://doi.org/10.18523/2617-7080i2018p11-14>
7. Abramov, V., Astafieva, M., Boiko, M., Bodnenko, D., Bushma, A., Vember, V., Hlushak, O., Zhylytsov, O., Ilich, L., Kobets, N., Kovaliuk, T., Kuchakovska, H., Lytvyn, O., Lytvyn, P., Mashkina, I., Morze, N., Nosenko, T., Proshkin, V., Radchenko, S., & Yaskevych, V. (2021). Theoretical and practical aspects of the use of mathematical methods and information technology in education and science. <https://doi.org/10.28925/9720213284km>