

Одеський національний університет імені І. І. Мечникова
Факультет математики, фізики та інформаційних технологій
Кафедра оптимального керування і економічної кібернетики

Кваліфікаційна робота

на здобуття ступеня вищої освіти «магістр»

**«Розв’язання часткової проблеми власних значень
симетричної матриці методом Ньютона»**

**«Solving of a partial eigenvalues problem of a symmetric
matrix by Newton’s method»**

Виконала: здобувачка денної форми навчання
спеціальності 113 Прикладна математика
Освітня програма «Прикладна математика»
Гук Аліна Григорівна

Керівник: к. ф.-м. н., доц. Вербіцький В. В. _____

Рецензент: к. ф.-м. н., доц. Васильєв О.Б. _____

Рекомендовано до захисту:

Захищено на засіданні ЕК № _____

Протокол засідання кафедри

Протокол № ____ від _____ 2023 р.

№ ____ від _____ 2023 р.

Оцінка _____ / _____ / _____

Завідувач кафедри

Голова ЕК

Одеса — 2023 р.

ЗМІСТ

Вступ	3
1 Постановка задачі	5
2 Симетрична проблема власних значень	6
3 Степеневий метод для симетричної проблеми власних значень	16
3.1 Степеневий метод та зворотна ітерація	16
3.2 Використання зсувів	20
4 Класичний метод Ньютона	23
5 Метод Ньютона для спектральної задачі	26
6 Метод ітерування підпростору	30
6.1 Реалізації	31
6.2 Удосконалення	35
6.2.1 Чебишове прискорення	35
6.2.2 Рандомізація	36
6.3 Збіжність	37
7 Обчислювальний експеримент	42
Висновки	44
Список літератури	45
Додаток А. Код програми	50

ВСТУП

Враховуючи відомість досить точного наближення до розв'язання рівняння $F(x) = 0$, метод Ньютона є ефективним засобом підвищення точності цього наближення. Багато тверджень про збіжність методу Ньютона взяті з результатів Л.В. Канторовича [?], який застосував метод Ньютона до нелінійних операторних рівнянь у банахових просторах.

Відомо багато спроб застосування методу Ньютона до спектральних задач. Без намагань повністю охопити цю історію, важливо відзначити деякі її етапи. Наприклад, у монографії Дж. Х. Уілкінсона [15] розглянуто використання методу Ньютона для знаходження коренів характеристичного рівняння $\det(A - \lambda I) = 0$, які визначають власні значення матриці A . Інший приклад - у монографії Фаддєєва Д.К. та Фаддєєвої В.М. [?], де метод Ньютона використовується для уточнення окремого власного значення та відповідного йому власного вектора, перша компонента якого відрізняється від інших тим, що не є зникаюче малою, так що порушення спільності можна вважати її рівної одиниці. Отримане нелінійне рівняння стосовно власного значення вирішується методом Ньютона. В статті В. Н. Кублановської [?] пропонується алгоритм знаходження комплексно-сполучених власних значень та власних векторів дійсної матриці в дійсній арифметиці. Для обчислення дійсної та уявної частин власного значення застосовується метод Ньютона для знаходження коренів нелінійних рівнянь. Важливо відзначити, що в усіх вищезгаданих випадках метод Ньютона застосовувався для скалярних нелінійних рівнянь з метою уточнення власного значення.

Інший підхід описано у монографії Л. Коллатца [?]. Задача про власні значення $Ax = \lambda Bx$ (A, B - задані матриці розмірністю $n \times n$) зводиться до пошуку коренів нелінійної системи з $n + 1$ рівнянь:

$$\begin{cases} Ax - \lambda Bx = 0, & x_n - 1 = 0. \end{cases} \quad (0.1)$$

Стверджується, що метод Ньютона є дуже ефективним для пошуку коренів системи (0.1). Для доведення збіжності методу Ньютона для системи (0.1) запропоновано використовувати теорему про збіжність методу Ньютона для нелінійного операторного рівняння в банаховому просторі. Проте довести

виконання умов цієї теореми практично складно. Доведення збіжності методу Ньютона наведено лише для числового прикладу з матрицями розмірності 3×3 . При цьому при доведенні використовуються оцінки, що було отримано на підставі величин, знайдених у процесі обчислень. Інших доведень збіжності методу Ньютона для системи виду (0.1) автори не виявили. Нами запропоновано просте доведення збіжності методу Ньютона для системи виду (0.1), яке базується на відомій теоремі про збіжність методу Ньютона для систем нелінійних рівнянь.

Об'єктом дослідження роботи є спектральна задача для симетричної матриці.

Предметом дослідження роботи є числові методи обчислення власних значень та векторів симетричної матриці.

Метою роботи є дослідження можливості використання методу Ньютона для обчислення власних значень та векторів симетричної матриці.

Для досягнення мети роботи необхідно: модифікувати метод Ньютона для обчислення власних значень та векторів симетричної матриці; довести збіжність методу; порівняти запропонований метод з відомими методами; провести обчислювальний експеримент.

РОЗДІЛ 1

ПОСТАНОВКА ЗАДАЧІ

Розглянемо наступну задачу на власні значення

$$Ax = \lambda x, \quad \text{де}$$

$$A = A^T \in R^n$$

$\{x^0, \lambda^0\}$ — початкове наближення до деякої власної пари з простим власним значенням.

Метод зворотна ітерації з відношенням Релея застосовується найчастіше для уточнення власної пари.

Метод ітерування підпростору може бути ефективним для роботи з матрицями великих розмірностей або у випадках, коли потрібно знайти лише кілька власних пар.

Потрібно розглянути можливість застосування методу Ньютона для уточнення власної пари.

РОЗДІЛ 2

СИМЕТРИЧНА ПРОБЛЕМА ВЛАСНИХ ЗНАЧЕНЬ

Власні значення матриці $A \in C^{n \times n}$ — це n корені її *характеристичного багаточлена*

$$p_n(\lambda) = \det(A - \lambda I). \quad (2.1)$$

Спектр — множина усіх коренів. Позначається $\lambda(A)$.

Власний вектор — ненульовий вектор $x \in C^n$, який при $\lambda \in \lambda(A)$ задовольняє рівнянню

$$Ax = \lambda x.$$

Алгебраїчна кратність власного значення λ — це кратність кореня λ характеристичного багаточлена (2.1)

Геометрична кратність власного значення λ — кількість лінійно незалежних власних векторів, що йому відповідають.

Лема 2.1. (Лема Гершгоріна). *Усі власні значення комплексної матриці $A \in C^{n \times n}$ знаходяться в об'єднанні кіл з центрами в точках a_{ii} і радіусами*

$$r_{ii} = \sum_{j=1 \atop (j \neq i)}^n |a_{ij}|, \quad i = \overline{1, n}.$$

Доведення. Нехай $\{\lambda, x\}$ — власна пара матриці A і

$$|x_i| = \max_j |x_j|.$$

Тоді

$$a_{i1}x_1 + \cdots + a_{in}x_n = \lambda x_i,$$

або

$$(\lambda - a_{ii})x_i = a_{i1}x_1 + \cdots + a_{i,i-1}x_{i-1} + a_{i,i+1}x_{i+1} + \cdots + a_{in}x_n.$$

Таким чином

$$|\lambda - a_{ii}| \leq \sum_{j=1 \atop (j \neq i)}^n |a_{ij}|, \quad i = \overline{1, n}.$$

□

Зазвичай, лема Гершгоріна надає дещо завищені оцінки меж спектру.

Власний вектор визначає одновимірний підпростір, інваріантний по відношенню до множення на матрицю A зліва. Підпростір $S \subseteq C^n$, що має властивість

$$x \in S \Rightarrow Ax \in S,$$

загалом, називають інваріантним підпростором матриці A .

Лема 2.2. *Нехай $X \in C^{n \times k}$, $B \in C^{k \times k}$ ($k \leq n$). Якщо*

$$AX = XB,$$

тоді підпростір $S = \text{range}(X)$ інваріантний. До того ж $\lambda(B) \subseteq \lambda(A)$, якщо матриця X має повний стовпцевий ранг.

Доведення. Якщо $x \in S = \text{range}(X)$, то існує такий вектор $y \in C^k$, що $x = Xy$. То

$$Ax = AXy = XBy = Xz \in S = \text{range}(X).$$

Нехай $\lambda \in \lambda(B)$. То існує такий вектор $y \neq 0$, що $By = \lambda y$. Тоді

$$AXy = XBy = \lambda Xy$$

і так як матриця X повного стовпцевого ранга, вектор Xy не дорівнює нулю. Отримуємо, що $\{\lambda, Xy\}$ буде власною парою матриці A . □

Важливо зазначити, що, коли квадратна матриця X з леми, що ми розглянули до цього, є невиродженою, то $\lambda(B) = \lambda(A)$, а матриці A та $B = X^{-1}AX$ в такому випадку подібні. Тоді X є перетворенням подібності. Отже, перетворення подібності зберігає власні значення матриці.

Існує багато алгоритмів для обчислення власних значень, в яких відбувається розбиття вихідної задачі на деяку кількість менших за розміром

задач. Цей результат є основою для подальших перетворень.

Лема 2.3. *Якщо матрицю $T \in C^{n \times n}$ можна представити у наступному вигляді*

$$T = \begin{bmatrix} T_{11} & T_{12} \\ 0 & T_{22} \end{bmatrix}$$

тоді $\lambda(T) = \lambda(T_{11}) \cup \lambda(T_{22})$.

Доведення. Нехай

$$Tx = \begin{bmatrix} T_{11} & T_{12} \\ 0 & T_{22} \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} = \lambda \begin{bmatrix} x_1 \\ x_2 \end{bmatrix},$$

де $x_1 \in C^p$ і $x_2 \in C^q$. Якщо $x_2 \neq 0$, тоді $T_{22}x_2 = \lambda x_2$ і $\lambda \in \lambda(T_{22})$. Якщо $x_2 = 0$, тоді $T_{11}x_1 = \lambda x_1$ та $\lambda \in \lambda(T_{11})$. Таким чином, $\lambda(T) \in \lambda(T_{11}) \cup \lambda(T_{22})$. Однак, так як в множинах $\lambda(T)$ та $\lambda(T_{11}) \cup \lambda(T_{22})$ міститься однакова кількість елементів, то ці дві множини є рівними. $\square$

Лема 2.4. *Якщо матриці $A \in C^{n \times n}$, $X \in C^{n \times p}$, $B \in C^{p \times p}$ ($p < n$) задовольняють умовам*

$$AX = XB, \quad \text{rank}(X) = p, \quad (2.2)$$

тоді існує така унітарна матриця $Q \in C^{n \times n}$, що

$$Q^T A Q = T = \begin{bmatrix} T_{11} & T_{12} \\ 0 & T_{22} \end{bmatrix},$$

де $\lambda(T_{11}) = \lambda(A) \cap \lambda(B)$.

Доведення. Нехай

$$X = Q \begin{bmatrix} R \\ 0 \end{bmatrix}, \quad Q \in C^{n \times n}, R \in C^{p \times p}$$

це QR -розклад матриці X . Якщо ми підставимо його в (2.2) та перетворимо цей вираз, отримаємо

$$\begin{bmatrix} T_{11} & T_{12} \\ T_{21} & T_{22} \end{bmatrix} \begin{bmatrix} R \\ 0 \end{bmatrix} = \begin{bmatrix} R \\ 0 \end{bmatrix} B,$$

де

$$Q^T A Q = \begin{bmatrix} T_{11} & T_{12} \\ T_{21} & T_{22} \end{bmatrix}$$

З невинродженості матриці R та рівнянь $T_{21}R = 0$, $T_{11}R = RB$ випливає, що $T_{21} = 0$ та $\lambda(T_{11}) = \lambda(B)$. Застосуємо лему 2.3, згідно з якою $\lambda(A) = \lambda(T) = \lambda(T_{11}) \cup \lambda(T_{22})$. Таким чином, лему доведено. $\square$

Лема 2.4 стверджує, що будь-яку квадратну матрицю можна привести до блочно-трикутної форми, якщо відомо хоча б один з її інваріантних підпросторів.

Теорема 2.1. (Розклад Шура). *Для довільної матриці $A \in C^{n \times n}$ існує унітарна матриця $Q \in C^{n \times n}$ така, що*

$$Q^H A Q = T = D + N,$$

де $D = \text{diag}(\lambda_1, \dots, \lambda_n)$ та $N \in C^{n \times n}$ є строго верхньою трикутною матрицею. Зазначимо, що матрицю Q можна обрати так, що власні значення λ_i будуть розташовані у довільному заданому порядку.

Доведення. Скористаємося індукцією для доведення теореми. Теорема є вірною при $n = 1$. Припускаємо, що теорема є вірною і для всіх матриць порядку $n - 1$ або менше. Згідно з лемою 2.4 (при $B = (\lambda)$), якщо $Ax = \lambda x$ ($x \neq 0$), тоді існує унітарна матриця U така, що

$$U^H A U = \begin{bmatrix} \lambda & w^H \\ 0 & C \end{bmatrix}.$$

Враховуючи припущення індукції, існує така унітарна матриця $\tilde{U} \in C^{(n-1) \times (n-1)}$, що матриця $\tilde{U}^H C \tilde{U}$ є строго верхньою трикутною. Отже, якщо $Q = U \text{diag}(1, \tilde{U})$, то матриця $Q^H A Q$ також є строго верхньою трикутною. $\square$

Теорема 2.2. (Розклад Жордана). *Якщо $A \in C^{n \times n}$, тоді існує невроджена матриця $X \in C^{n \times n}$ така, що*

$$X^{-1} A X = \text{diag}(J_1, \dots, J_k),$$

де

$$J_i = \begin{bmatrix} \lambda_i & 1 & & \\ & \ddots & \ddots & \\ & & \ddots & 1 \\ & & & \lambda_i \end{bmatrix} \in C^{m_i \times m_i}$$

і $m_1 + \dots + m_k = n$.

Доведення. Див., наприклад, [?]. $\square$

Блоки J_i називаються жордановими. Кількість і розміри жорданових блоків, пов'язаних з кожним власним значенням, визначаються однозначно. Проте їхнє розташування вздовж діагоналі є неоднозначним.

Дійсний розклад Шура представляє собою блокову верхню трикутну матрицю з діагональними блоками розміром 1×1 та 2×2 .

Теорема 2.3. (Дійсний розклад Шура). *Для будь-якої дійсної матриці $A \in R^{n \times n}$ існує ортогональна матриця $Q \in R^{n \times n}$ така, що*

$$Q^T A Q = \begin{bmatrix} R_{11} & R_{11} & \dots & R_{1m} \\ 0 & R_{22} & \dots & R_{2m} \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & R_{mm} \end{bmatrix}, \quad (2.3)$$

де кожен блок R_{ii} є матрицею розмірності 1×1 або 2×2 і має комплексно-спряжені власні значення.

Доведення. Власні значення матриці A , що є комплексно-спряженими парами, повинні входити в $\lambda(A)$ комплексно-спряженими парами, оскільки характеристичний багаточлен $\det(A - \lambda I)$ має дійсні коефіцієнти. Нехай k — це кількість комплексно-спряжених пар в $\lambda(A)$. Якщо $k = 0$, тоді теорема доводиться аналогічно доведенню теореми 2.1, тобто із застосуванням леми 2.4, зазначивши, що коефіцієнти та власні значення матриці A є дійсними.

Припустимо $k \geq 1$. Тоді існують такі вектори y та z ($z \neq 0$) в R^n при $\lambda = \gamma + i\mu \in \lambda(A)$ та $\mu \neq 0$, що

$$A(y + iz) = (\gamma + i\mu)(y + iz).$$

Якщо дійсні та уявні частини прирівняти, то отримаємо

$$A[y, z] = [y, z] \begin{bmatrix} \gamma & \mu \\ -\mu & \gamma \end{bmatrix}.$$

Таким чином, вектори y та z утворюють дійсний двовимірний інваріантний підпростір матриці A . З леми 2.4 випливає, що існує ортогональна матриця $U \in R^{n \times n}$ така, що

$$U^T A U = \begin{bmatrix} T_{11} & T_{12} \\ 0 & T_{22} \end{bmatrix},$$

де $\lambda(T_{11}) = \{\lambda, \bar{\lambda}\}$. Враховуючи припущення індукції, можна знайти ортогональну матрицю $\tilde{U}$ таку, що матриця $\tilde{U}^T T_{22} \tilde{U}$ матиме необхідну структуру. Якщо взяти $Q = U \text{diag}(I_2, \tilde{U})$, то ми отримуємо необхідний нам розклад. $\square$

Теорема 2.3 демонструє, що будь-яка дійсна матриця може бути ортогонально подібною до певної верхньої майже трикутної матриці. Дійсні та уявні частини комплексних власних значень можна легко визначити з діагональних блоків розмірністю 2×2 .

Теорема 2.4. (Теорема про спектральний розклад). Для будь-якої дійсної симетричної матриці $A \in R^{n \times n}$ існує ортогональна матриця $Q \in R^{n \times n}$ така, що

$$Q^T A Q = \Lambda = \text{diag}(\lambda_1, \dots, \lambda_n). \quad (2.4)$$

Доведення. Зауважимо, що так як матриця A є симетричною, тоді в розкладі (2.3) згідно з теоремою 2.3 матриця $Q^T A Q$ теж є симетричною, та, отже, діагональною. Таким чином, з теореми 2.3 випливає доведення теореми 2.4. $\square$

З теореми 2.4 випливає, що у дійсної симетричної $n \times n$ -матриці є n дійсних власних значень та n відповідних їм власних векторів. Ці вектори можна використовувати для побудови ортонормованого базису у просторі R^n .

Якщо (2.4) помножити зліва на Q , а справа на Q^T , тоді отримаємо

$$A = Q \Lambda Q^T = \lambda_1 q_1 q_1^T + \lambda_2 q_2 q_2^T + \dots + \lambda_n q_n q_n^T = \sum_{i=1}^n \lambda_i H_i, \quad (2.5)$$

де $H_i = q_i q_i^T$ є ортогональним проектором на одинірний власний підпростір власного вектора q_i . У випадку, коли власному значенню λ відповідає кілька власних векторів (тобто власне значення кратне), то ці вектори визначаються неоднозначно. Однак підпростір, який розтягнутий на ці власні вектори, визначається однозначно. Таким чином, для кратного власного значення λ ми визначаємо ортогональний проектор на його власний підпростір таким чином:

$$H_\lambda = \sum_{\lambda_j = \lambda} H_j.$$

З (2.5) отримаємо розклад

$$A = \sum_{j=1}^m \lambda_j H_{\lambda_j}, \quad (2.6)$$

де λ_j ($j = \overline{1, m}$) різні між собою. Розклад, що ми отримали, а саме (2.6) являє собою ніщо інше як спектральний розклад дійсної симетричної матриці A . у випадку симетричної матриці алгебраїчна та геометрична кратності

власного значення рівні одна одній.

Оскільки, згідно з попередньою теоремою, всі власні значення дійсні симетричної матриці дійсні, то з леми Гершгоріна можна вивести таке слідство.

Наслідок 2.1. *Власні значення дійсної симетричної матриці $A \in R^{n \times n}$ належать об'єднанню інтервалів*

$$[a_{ii} - r_{ii}, a_{ii} + r_{ii}], \text{ де } r_{ii} = \sum_{j=1 \atop (j \neq i)}^n |a_{ij}|, \quad i = \overline{1, n}.$$

Приклад 2.1. Матриця

$$A = \begin{bmatrix} 2 & 1 & 1 & 0 \\ 1 & 2 & 0 & 1 \\ 1 & 0 & 2 & 1 \\ 0 & 1 & 1 & 2 \end{bmatrix}$$

має власні значення $\lambda_1 = 0$, $\lambda_2 = \lambda_3 = 2$, $\lambda_4 = 4$. Власному значенню λ_1 відповідають власний вектор $q_1 = 1/2[1, -1, -1, 1]^T$ та ортогональний проектор

$$H_{\lambda_1} = q_1 q_1^T = \frac{1}{4} \begin{bmatrix} 1 & -1 & -1 & 1 \\ -1 & 1 & 1 & -1 \\ -1 & 1 & 1 & -1 \\ 1 & -1 & -1 & 1 \end{bmatrix}.$$

Власному значенню λ_4 відповідає власний вектор

$$q_4 = 1/2[1, 1, 1, 1]^T$$

та ортогональний проектор

$$H_{\lambda_4} = q_4 q_4^T = \frac{1}{4} \begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 \end{bmatrix}.$$

Власному значенню $\lambda = 2$ (кратності 2) відповідають два власного вектора. Їх можна визначити по-різному. Наприклад,

$$q_2 = 1/2[1, -1, 1, -1]^T \quad \text{и} \quad q_3 = 1/2[1, 1, -1, -1]^T$$

або

$$q_2 = 1/\sqrt{2}[1, 0, 0, -1]^T \quad \text{и} \quad q_3 = 1/\sqrt{2}[0, 1, -1, 0]^T.$$

Легко перевірити, що в обох випадках ортогональний проектор на власний підпростір власного значення $\lambda = \lambda_2 = \lambda_3 = 2$ визначається однозначно

$$H_\lambda = q_2 q_2^T + q_3 q_3^T = \frac{1}{2} \begin{bmatrix} 1 & 0 & 0 & -1 \\ 0 & 1 & -1 & 0 \\ 0 & -1 & 1 & 0 \\ -1 & 0 & 0 & 1 \end{bmatrix}.$$

Доведемо наступну лему.

Лема 2.5. Для симетричної матриці $A \in R^{n \times n}$ та довільного вектора $x \neq 0$

$$\min_{\sigma} \|Ax - \sigma x\|_2 = \|Ax - \rho(x)x\|_2,$$

де $\rho(x) = \frac{x^T A x}{x^T x}$ називається відношенням Релея матриці A .

Доведення. Дійсно, функція

$$f(\sigma) = \|Ax - \sigma x\|_2^2 = x^T A^2 x - 2\sigma x^T A x + \sigma^2 x^T x$$

досягає свого мінімуму в точці $\sigma = \frac{x^T A x}{x^T x}$

□

З леми можна зробити наступний висновок. Якщо $x \neq 0$ є наближенням до деякого власного вектора симетричної матриці A , то найкраще наближення до відповідного власного значення, яке можна обчислити знаючи вектор x , дає відношення Релея $\rho(x)$.

Нехай $\{q_i\}_{i=1}^n$ — ортонормований базис R^n складений із власних векторів матриці A . Тоді будь-який вектор $x \in R^n$ можна представити у вигляді

$$x = \alpha_1 q_1 + \alpha_2 q_2 + \dots + \alpha_n q_n.$$

Таким чином,

$$\rho(x) = \frac{x^T A x}{x^T x} = \frac{\sum_{i=1}^n \lambda_i \alpha_i^2}{\sum_{i=1}^n \alpha_i^2}.$$

На основі цього отримуємо

$$\lambda_{\min} \leq \frac{x^T A x}{x^T x} \leq \lambda_{\max} \quad \forall x \in R^n \setminus 0.$$

РОЗДІЛ 3

СТЕПЕНЕВИЙ МЕТОД ДЛЯ СИМЕТРИЧНОЇ ПРОБЛЕМИ ВЛАСНИХ ЗНАЧЕНЬ

Степеневий метод та зворотна ітерація

Припустимо, найбільше за модулем власне значення дійсної симетричної матриці $A \in R^{n \times n}$ є відокремленим, а саме,

$$|\lambda_1| > |\lambda_2| \geq \dots \geq |\lambda_n|.$$

В такому разі степеневий метод дозволяє знайти максимальне по модулю власне значення λ_1 та власний вектор q_1 , що йому відповідає.

Запустимо подальший ітераційний процес. Нехай y_0 є початковим наближенням до власного вектора q_1 . За теоремою 2.4, що стосується спектрального розкладу симетричної матриці, власні вектори матриці A утворюють ортонормований базис простору $R^{n \times n}$. Тоді якщо розкладемо y_0 по цьому базису,

$$y_0 = \sum_{i=1}^n \alpha_i q_i, \tag{3.1}$$

де $\alpha_i = q_i^T y_0$. Припустимо, що $\alpha_1 \neq 0$. Далі пронормуємо вектор y_0 ,

$$x_0 = \frac{y_0}{\|y_0\|_2}.$$

Якщо помножити матрицю A на вектор x_0 , тоді

$$y_1 = Ax_0 = \frac{\sum_{i=1}^n \alpha_i \lambda_i q_i}{\left\| \sum_{i=1}^n \alpha_i q_i \right\|_2}.$$

Пронормуємо вектор y_1 ,

$$x_1 = \frac{y_1}{\|y_1\|_2} = \frac{\sum_{i=1}^n \alpha_i \lambda_i q_i}{\left\| \sum_{i=1}^n \alpha_i \lambda_i q_i \right\|_2}.$$

Після проведення k ітерацій маємо

$$x_k = \frac{\sum_{i=1}^n \alpha_i \lambda_i^k q_i}{\left\| \sum_{i=1}^n \alpha_i \lambda_i^k q_i \right\|_2}, \quad (3.2)$$

$$y_{k+1} = Ax_k = \frac{\sum_{i=1}^n \alpha_i \lambda_i^{k+1} q_i}{\left\| \sum_{i=1}^n \alpha_i \lambda_i^k q_i \right\|_2}. \quad (3.3)$$

Припустимо

$$\lambda^{(k+1)} = y_{k+1}^T x_k.$$

Покажемо, що $\lambda^{(k+1)} \rightarrow \lambda_1$ при $k \rightarrow \infty$. »з (3.2) та (3.3) маємо

$$\begin{aligned} \lambda^{(k+1)} &= y_{k+1}^T x_k = \frac{\left(\sum_{i=1}^n \alpha_i \lambda_i^{k+1} q_i \right)^T \left(\sum_{i=1}^n \alpha_i \lambda_i^k q_i \right)}{\left\| \sum_{i=1}^n \alpha_i \lambda_i^k q_i \right\|_2^2} = \frac{\sum_{i=1}^n \alpha_i^2 \lambda_i^{2k+1}}{\sum_{i=1}^n \alpha_i^2 \lambda_i^{2k}} = \\ &= \frac{\lambda_1 + \frac{\alpha_2^2 \lambda_2}{\alpha_1^2} \left(\frac{\lambda_2}{\lambda_1} \right)^{2k} + \dots + \frac{\alpha_n^2 \lambda_n}{\alpha_1^2} \left(\frac{\lambda_n}{\lambda_1} \right)^{2k}}{1 + \frac{\alpha_2^2}{\alpha_1^2} \left(\frac{\lambda_2}{\lambda_1} \right)^{2k} + \dots + \frac{\alpha_n^2}{\alpha_1^2} \left(\frac{\lambda_n}{\lambda_1} \right)^{2k}} = \frac{\lambda_1 + O\left(\left(\frac{\lambda_2}{\lambda_1}\right)^{2k}\right)}{1 + O\left(\left(\frac{\lambda_2}{\lambda_1}\right)^{2k}\right)}. \end{aligned}$$

Виходячи з цього

$$\lambda^{(k+1)} - \lambda_1 = O\left(\left(\frac{\lambda_2}{\lambda_1}\right)^{2k}\right). \quad (3.4)$$

Тепер продемонструємо, що $x_k \rightarrow q_1$ при $k \rightarrow \infty$. І втім,

$$\begin{aligned} x_k &= \frac{\sum_{i=1}^n \alpha_i \lambda_i^k q_i}{\left\| \sum_{i=1}^n \alpha_i \lambda_i^k q_i \right\|_2} = \frac{\alpha_1 \lambda_1^k q_1 + \alpha_2 \lambda_2^k q_2 + \dots + \alpha_n \lambda_n^k q_n}{|\alpha_1| |\lambda_1|^k \sqrt{1 + O\left(\left(\frac{\lambda_2}{\lambda_1}\right)^{2k}\right)}} = \\ &= \frac{\pm q_1 + O\left(\left(\frac{\lambda_2}{\lambda_1}\right)^k\right)}{\sqrt{1 + O\left(\left(\frac{\lambda_2}{\lambda_1}\right)^{2k}\right)}}. \end{aligned}$$

Виходячи з цього

$$x_k - q_1 = O\left(\left(\frac{\lambda_2}{\lambda_1}\right)^k\right). \quad (3.5)$$

Зазначимо, що темп збіжності власного значення більший у два рази, аніж темп збіжності власного вектора.

Напишемо алгоритм степеневого методу.

Степеновий метод

1. $x_0 = y_0 / \|y_0\|_2$
2. *For* $k = 0, 1, \dots$ *Until Convergence Do*
3. $y_{k+1} = Ax_k$
4. $\lambda^{(k+1)} = y_{k+1}^T x_k$
5. $x_{k+1} = y_{k+1} / \|y_{k+1}\|_2$
6. *End Do*

Як перевірку на збіжність може служити одна з умов,

$$\left| \lambda^{(k+1)} - \lambda^{(k)} \right| \leq \varepsilon \left| \lambda^{(k+1)} \right|, \quad (3.6)$$

чи

$$\left| \max_{(x_k)_i \neq 0} \frac{(y_{k+1})_i}{(x_k)_i} - \min_{(x_k)_i \neq 0} \frac{(y_{k+1})_i}{(x_k)_i} \right| \leq \varepsilon \left| \max_{(x_k)_i \neq 0} \frac{(y_{k+1})_i}{(x_k)_i} \right|. \quad (3.7)$$

Зазначимо, щоо крок 3. алгоритму 3.1 вимагає $O(n^2)$ арифметичних дій, оді як всі інші кроки вимагають $O(n)$ арифметичних дій.

Тепер розглянемо наступний метод - метод зворотної ітерації. Припустимо, що власне значення дійсної симетричної матриці $A \in R^{n \times n}$, що є найменшим за модулем, є відокремленим, а саме,

$$|\lambda_1| \geq \dots \geq |\lambda_{n-1}| > |\lambda_n|. \quad (3.8)$$

Зазначимо, якщо $\{\lambda, x\}$ є власною парою матриці A , тоді $\{1/\lambda, x\}$ є власною парою матриці зворотної до A , тобто A^{-1} . Розумно вважати $1/\lambda_n$ власним значенням матриці A^{-1} найбільшим за модулем. В такому разі для визначення $1/\lambda_n$ слід використовувати степеневий метод для матриці A^{-1} . В такому разі при $k \rightarrow \infty$

$$\lambda^{(k+1)} \rightarrow \frac{1}{\lambda_n}, \quad x_k \rightarrow q_n.$$

Отже, застосування зворотної ітерації дозволяє визначити найменше за модулем власне значення і відповідний йому власний вектор матриці, коли воно є відокремленим від інших.

Напишемо алгоритм зворотної ітерації.

Зворотня ітерація

1. $x_0 = y_0 / \|y_0\|_2$
2. *For* $k = 0, 1, \dots$ *Until Convergence Do*
3. $y_{k+1} = A^{-1}x_k$
4. $\lambda^{(k+1)} = y_{k+1}^T x_k$
5. $x_{k+1} = y_{k+1} / \|y_{k+1}\|_2$
6. *End Do*

Крок 3. алгоритму зворотної ітерації можна звести до знаходження розв'язку системи $Ay_{k+1} = x_k$. Треба провести $O(n^3)$ арифметичних дій для того, щоб вирішити систему вказану вище. Але важливо зазначити, що треба провести лише $O(n^2)$ арифметичних процесів, коли вже відомий якийсь розклад матриці A . Скажімо розклад LU . Інші кроки цього алгоритму потребують тільки $O(n)$ арифметичних процесів.

Зауваження 3.1. Відновлюючи збіжність степеневому методу, передбачалося (див. (3.1)), що α_1 — коефіцієнт у розкладі за власними векторами

початкового наближення y_0 — не дорівнює нулю. У випадку, коли це не вірно, то, у результаті округлень при здійсненні арифметичних дій, на деякій k коефіцієнт α_1 у розкладі y_k не дорівнюватиме нулеві.

Використання зсувів

Нехай $A \in R^{n \times n}$ і $\sigma \in R$. Коли $\{\lambda, x\}$ є власною парою матриці A , тоді $\{\lambda - \sigma, x\}$ — власна пара матриці $A - \sigma I$. Величина σ є дійсним числом і несе назву зсува.

На початку слід розглянути застосування зсуву для збільшення швидкості збіжності зворотної ітерації. Так як зворотна ітерація — це метод степеневого виду для оберненої матриці, тоді

$$\lambda^{(k+1)} - \frac{1}{\lambda_n} = O \left(\left(\frac{\frac{1}{\lambda_{n-1}}}{\frac{1}{\lambda_n}} \right)^{2k} \right) = O \left(\left(\frac{\lambda_n}{\lambda_{n-1}} \right)^{2k} \right).$$

Таким чином, маємо, що швидкість збіжності пов'язана з величиною $|\frac{\lambda_n}{\lambda_{n-1}}|$, а саме наскільки невеликою вона є. Ця величина залежить від відстані між $|\lambda_{n-1}|$ та $|\lambda_n|$, а саме, від відокремлення найменшого за модулем власного значення і від малості $|\lambda_n|$. Зсув σ же визначається в такому разі так, щоб власне значення $\tilde{\lambda}_n = \lambda_n - \sigma$ матриці $A - \sigma I$, найменше по модулю) прагнуло до нуля. Так, для збільшення швидкості збіжності методу зсув σ має бути

В такому разі зсув σ обирають так, щоб найменше за модулем власне значення $\tilde{\lambda}_n = \lambda_n - \sigma$ матриці $A - \sigma I$ було якомога ближче до нуля. Таким чином, для збільшення швидкості збіжності зворотної ітерації зсув σ треба вибирати скільки можливо наближено до шуканого власного значення λ_n . Якщо $\tilde{x}$ є наближенням до якогось власного вектора симетричної матриці, тоді згідно з лемою 2.5 найкраще наближення до відповідного власного значення отримується за допомогою відношення Релея

$$\rho(\tilde{x}) = \frac{\tilde{x}^T A \tilde{x}}{\tilde{x}^T \tilde{x}}.$$

Це служить основою для подальшого алгоритму.

Алгоритм 3.1. Зворотня ітерація зі зсувом Релея

1. $x_0 = y_0 / \|y_0\|_2$
2. *For* $k = 0, 1, \dots$ *Until Convergence Do*
3. $\sigma_{k+1} = x_k^T A x_k$
4. $y_{k+1} = (A - \sigma_{k+1} I)^{-1} x_k$
5. $x_{k+1} = y_{k+1} / \|y_{k+1}\|_2$
6. *End Do*

У ролі критерію збіжності можна застосувати умову (3.7).

Відмітимо, що крок 3. алгоритму вимагає $O(n^3)$ арифметичних процесів, оскільки кожен раз потрібно вирішувати систему лінійних рівнянь матриці з новою матрицею.

Розглянемо зараз застосування зсуву для обчислення власних значень матриці зворотною ітерацією, які відрізняються за модулем від максимальних (мінімальних). Зрозуміло, що степеневий метод (зворотня ітерація) в процесі обчислення мінімального за модулем власного значення матриці $A - \sigma I$, обчислюватиме насправді власне значення матриці A , що є найбільшим до зсуву.

Встановивши межі спектра матриці A (скажімо, за лемою Гершгорина), зсув σ підбирається так, що степеневий метод чи зворотня ітерація обчислюють насправді найбільше або найменше власне значення матриці A . І справді (див. рис. 6.1), коли $\sigma \leq \lambda_{\min}(A)$, тоді усі власні значення матриці $A - \sigma I$ є додатними, найменшому власному значенню матриці A співвідноситься найменше за модулем власне значення матриці $A - \sigma I$, а найбільшому власному значенню матриці A — найбільше за модулем власне значення матриці $A - \sigma I$. В разі, коли $\sigma \geq \lambda_{\max}(A)$, тоді усі власні значення матриці $A - \sigma I$ від'ємні.

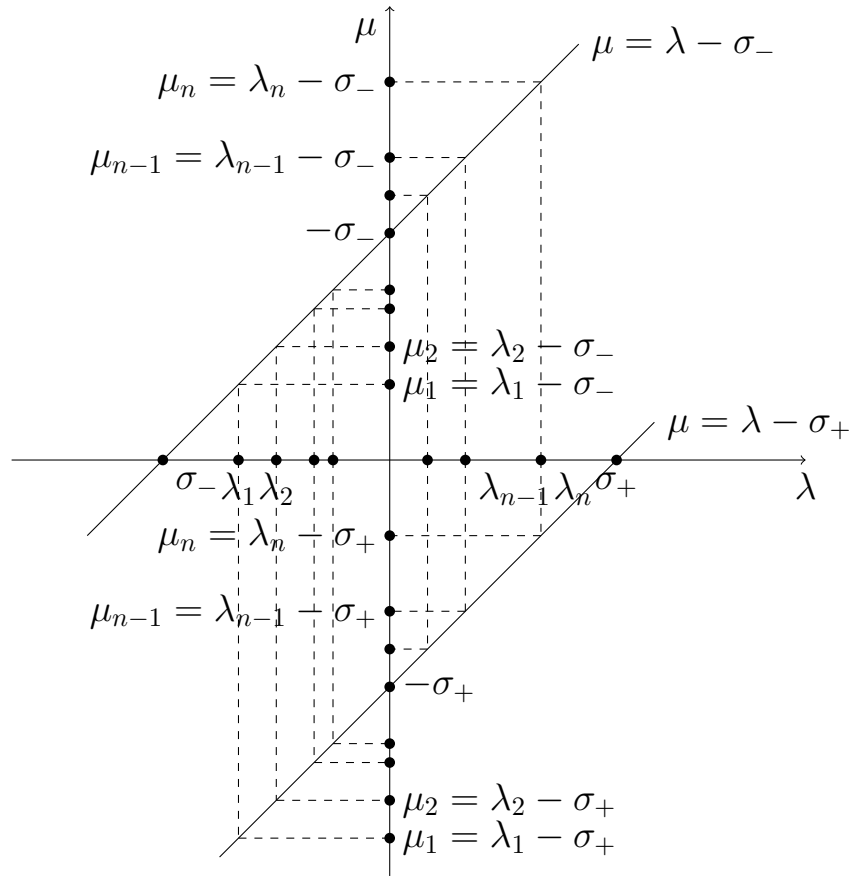


Рис. 3.1. Власні значення μ матриці $A - \sigma I$

РОЗДІЛ 4

КЛАСИЧНИЙ МЕТОД НЬЮТОНА

Розглянемо рівняння

$$F(x) = 0, \quad (4.1)$$

де $F : R^n \rightarrow R^n$ — гладке відображення. Для такого рівняння вводиться класичний метод Ньютона. Нехай $x^k \in R^n$ є поточним наближенням до шуканого кореню $\bar{x}$ рівняння (4.1). Поблизу x^k рівняння апроксимується лінеаризацією. Таке рівняння називається ітераційним рівнянням *методу Ньютона*.

$$F(x^k) + F'(x^k)(x - x^k) = 0. \quad (4.2)$$

Алгоритм 4.1. Оберемо $x^0 \in R^n$ та $k = 0$.

- 1) Знаходимо $x^{k+1} \in R^n$ як корінь рівняння (4.2)
- 2) Змінюємо крок k , збільшуючи його на 1. Переходимо до пункту 1.

Ітераційну схему методу Ньютона дуже часто представляють у вигляді:

$$x^{k+1} = x^k - (F'(x^k))^{-1} F(x^k), \quad k = 0, 1, \dots, \quad (4.3)$$

Звісно, очевидно, в реалізації методу абсолютно необов'язково обчислювати повне зворотнє відображення матриці $F'(x^k)$.

Теорема 4.1. *Припустимо, що відображення $F : R^n \rightarrow R^n$ диференційоване в деякому околі точки $\bar{x} \in R^n$, в той час як його похідна неперервна в цій точці. Нехай $\bar{x}$ — корінь рівняння (4.1), так що $\det F'(\bar{x}) \neq 0$. Таким чином будь-яке початкове наближення $x^0 \in R^n$, що є відносно наближене до $\bar{x}$, вказує траєкторію алгоритму 4.1, яка збігається до $\bar{x}$. Швидкість збіжності є лінійною, а якщо похідна F є неперервною по Ліпшицу в околі точки $\bar{x}$, то квадратичною.*

Доведення. Згідно з теоремою про малі збурення невинродженої матриці, існують такі окіли U точки $\bar{x}$ та число $M > 0$, що

$$\det F'(x) \neq 0, \quad \|(F'(x))^{-1}\| \leq M \quad \forall x \in U. \quad (4.4)$$

Окремо важливо відзначити, що для $x^k \in U$ рівняння (4.2) має єдиний розв'язок x^{k+1} , що означає, що ітерація алгоритму 1 з такої точки x^k однозначно встановлена.

Також, з умов (4.3), (4.4) та теореми про середнє значення можна обрати такий окіл U , що якщо $x^k \in U$, тоді

$$\begin{aligned} |x^{k+1} - \bar{x}| &= |x^k - \bar{x} - (F'(x^k) - F'(\bar{x}))| \leq \\ &\leq \|(F'(x^k))^{-1}\| |F(x^k) - F(\bar{x}) - F'(x^k)(x^k - \bar{x})| \leq \\ &\leq M \sup_{t \in [0,1]} \|F'(tx^k + (1-t)\bar{x}) - F'(x^k)\| |x^k - \bar{x}|. \end{aligned} \quad (4.5)$$

Важливо зазначити

$$\sup_{t \in [0,1]} \|F'(tx^k + (1-t)\bar{x}) - F'(x^k)\| \rightarrow 0 \quad (x^k \rightarrow \bar{x}).$$

Таким чином, з (4.5) випливає, що для будь-якого $q \in (0, 1)$ існує число $\delta > 0$ таке, що $B(\bar{x}, \delta) \subset U$, і якщо $x^k \in B(\bar{x}, \delta)$, тоді

$$|x^{k+1} - \bar{x}| \leq q|x^k - \bar{x}|;$$

Окремо відзначимо, що $x^{k+1} \in B(\bar{x}, \delta)$. Так, по-перше, будь-яке початкове наближення x^0 , достатньо близьке до $\bar{x}$, коректно визначає траєкторію алгоритму 1. По-друге, ця траєкторія збігається до $\bar{x}$ із надлінійною швидкістю.

В кінцевому результаті, коли похідна F є неперервною за Ліпшицем на U з константою $L > 0$, то з (4.6) при $x^k \in U$ отримуємо:

$$|x^{k+1} - \bar{x}| \leq ML|x^k - \bar{x}|^2, \quad (4.6)$$

що забезпечує квадратичну швидкість збіжності. □

Тут і в інших теоремах про методи типу методу Ньютона для квадратичної оцінки швидкості збіжності, замість безперервності похідної F за Ліпшицем, достатньо припустити виконання менш сильної умови

$$\|F'(x) - F'(\bar{x})\| \leq L|x - \bar{x}| \quad \forall x \in V,$$

де V є околом точки $\bar{x}$, а $L > 0$ є якоюсь константою. Але майже всюди далі в цьому контексті ми будемо вважати ліпшицеву безперервність похідної на околі розв'язку.

Іноді у виразі для $F'(\cdot)$ можна окреслити члени, які прагнуть до нуля в точці $\bar{x}$. Таке часто трапляється, коли відомий аналітичний вираз для $F'(\cdot)$. Логічно припустити, що такі члени рівні нулю, замінивши $F'(\cdot)$ відповідним апроксимуючим відображенням $F : \mathbb{R}^n \rightarrow \mathbb{R}^{n \times n}$ і розглядаючи *усічений метод Ньютона*, який можна представити у вигляді ітераційної схеми.

$$x^{k+1} = x^k - (F(x^k))^{-1} F(x^k), \quad k = 0, 1, \dots \quad (4.7)$$

Наслідок 4.1. *Нехай виконані умови теореми 4.1. Тоді для будь-якого відображення $F : \mathbb{R}^n \rightarrow \mathbb{R}^{n \times n}$ такого, що*

$$\|F(x) - F'(\bar{x})\| \rightarrow 0 \quad (x \rightarrow \bar{x}),$$

будь-яке початкове наближення $x^0 \in \mathbb{R}^n$, достатньо близьке до $\bar{x}$, вірно визначає траєкторію методу (4.7), яка збігається до $\bar{x}$. Швидкість збіжності є надлінійною, а коли похідна F є неперервною по Ліпшицу на околі точки $\bar{x}$ та існує окіл V точки $\bar{x}$ та число $N > 0$ такі, що

$$\|F(x) - F'(\bar{x})\| \leq N|x - \bar{x}| \quad \forall x \in V,$$

тоді є квадратичною.

[? , стр. 81]

РОЗДІЛ 5

МЕТОД НЬЮТОНА ДЛЯ СПЕКТРАЛЬНОЇ ЗАДАЧІ

Власні значення та відповідні власні вектори симетричної матриці $A \in \mathbb{R}^{n \times n}$ є розв'язками представленої нижче нелінійної системи

$$\begin{cases} Ax - \lambda x = 0, \\ \frac{1}{2}(1 - x^T x) = 0. \end{cases} \quad (5.1)$$

Відмітимо, що останнє рівняння системи є умовою нормування власного вектора. Представимо систему у вигляді

$$F \left(\begin{bmatrix} x \\ \lambda \end{bmatrix} \right) = \begin{bmatrix} Ax - \lambda x \\ \frac{1}{2}(1 - x^T x) \end{bmatrix}. \quad (5.2)$$

Похідна відображення F досить просто знаходиться

$$F' \left(\begin{bmatrix} x \\ \lambda \end{bmatrix} \right) = \begin{bmatrix} A - \lambda I & -x \\ -x^T & 0 \end{bmatrix}. \quad (5.3)$$

Введемо слідуючий ітераційний процес:

$$\begin{bmatrix} x^{k+1} \\ \lambda^{k+1} \end{bmatrix} = \begin{bmatrix} x^k \\ \lambda^k \end{bmatrix} - \left(F' \left(\begin{bmatrix} x^k \\ \lambda^k \end{bmatrix} \right) \right)^{-1} F \left(\begin{bmatrix} x^k \\ \lambda^k \end{bmatrix} \right), \quad k = 0, 1, \dots \quad (5.4)$$

Теорема 5.1. *Нехай $\bar{\lambda}$ — просте власне значення, $\bar{x}$ — відповідний власний вектор дійсної симетричної матриці A . Тоді для будь-яких початкових наближень $[x^0, \lambda^0]^T$, достатньо близьких до $[\bar{x}, \bar{\lambda}]^T$, метод Ньютона визначає послідовність, яка збігається до $[\bar{x}, \bar{\lambda}]^T$ із квадратичною швидкістю.*

Доведення. Покажемо, що відображення F ітераційного процесу задовільняє всі умови теореми 5.1.

Дійсно, відображення F є неперервно диференційованим, а його похідна є неперервною по Ліпшицу в околі $[\bar{x}, \bar{\lambda}]^T$. Тепер доведемо, що

$$\det F' \left(\begin{bmatrix} \bar{x} \\ \bar{\lambda} \end{bmatrix} \right) \neq 0.$$

Показати, що система

$$\begin{bmatrix} A - \bar{\lambda}I & -\bar{x} \\ -\bar{x}^T & 0 \end{bmatrix} \begin{bmatrix} x \\ \lambda \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \end{bmatrix} \quad (5.5)$$

Якщо помножити перше рівняння системи зліва на x^T та врахувати останнє рівняння, отримаємо

$$x^T Ax - \bar{\lambda} x^T x = 0.$$

Припустимо, що $x \neq 0$, тоді

$$\bar{\lambda} = \frac{x^T Ax}{x^T x}$$

та $x = \alpha \bar{x}$ ($\alpha \neq 0$), так як $\bar{\lambda}$ є простим власним значенням. Проте тоді з останнього рівняння системи отримаємо $\alpha \bar{x}^T \bar{x} = 0$, це є неможливим. Тоді $x = 0$. Враховуючи це, перше рівняння системи матиме вигляд

$$-\lambda \bar{x} = 0.$$

Це вказує на те, що $\lambda = 0$. Теорему доведено. $\square$

Введемо позначення λ^0 та x^0 для деяких наближених значень до власного числа $\bar{\lambda}$ та відповідного власного вектора $\bar{x}$ симетричної матриці A . Отже, k -й крок методу Ньютона може бути записаний наступним чином:

1) шукаємо корінь $[y^k, \mu^k]^T$ системи

$$\begin{bmatrix} A - \lambda^k I & -x^k \\ -(x^k)^T & 0 \end{bmatrix} \begin{bmatrix} y^k \\ \mu^k \end{bmatrix} = \begin{bmatrix} Ax^k - \lambda^k x^k \\ \frac{1}{2}(1 - (x^k)^T x^k) \end{bmatrix}; \quad (5.6)$$

2) вважаємо

$$\begin{cases} x^{k+1} = x^k - y^k, \\ \lambda^{k+1} = \lambda^k - \mu^k. \end{cases}$$

Відзначимо деякі особливості методу Ньютона. Права частина рівняння містить нев'язку $r^k = (A - \lambda^k)x^k$, яка є найефективнішим показником точності (λ^k, x^k) як власної пари матриці A , так і нев'язки рівняння нормування власного вектора. Таким чином, умови завершення ітераційного процесу можна встановлювати на основі цієї нев'язки:

$$\|r^k\| \leq \epsilon, \quad (5.7)$$

де ϵ — необхідна точність обчислень.

Ми порівняти метод Ньютона та метод зворотної ітерації з відношенням Релея за обчислювальними витратами. Крок k другого методу виглядає наступним чином

1. $\rho^k = (x^k)^T A x^k$,
2. $(A - \rho^k)y^{k+1} = x^k$,
3. $x^{k+1} = y^{k+1} / \|y^{k+1}\|_2$.

На кожному етапі обчислень головні витрати методів пов'язані із розв'язанням системи, матриця якої маняється за допомогою зсуву λ^k або ρ^k . У порівнянні з зворотною ітерацією із відношенням Релея, метод Ньютона може виявитися більш ефективним у випадках, коли власні значення обчислені із високою точністю. Якщо власне значення обчислено точно, то матриця $A - \rho^k I$ може стати виродженою в машинній арифметиці, що призводить до припинення обчислень. Проте можливість того, що відповідний власний вектор ще не обчислений з необхідною точністю, може вимагати продовження обчислень.

Як вже вказано вище, матриця системи завжди залишається невиродженою, навіть коли λ^k збігається до власного значення. Таким чином, застосування методу Ньютона можна продовжити для досягнення необхідної точності власного вектора, навіть якщо власне значення на даний момент точно обчислено.

РОЗДІЛ 6

МЕТОД ІТЕРУВАННЯ ПІДПРОСТОРУ

Ітерування підпростору — це пряме узагальнення степенного метода та методу зворотної ітерації. Якщо задано матрицю A і підпростір L з E^n , то нескладно визначити новий підпростір

$$AL \equiv \{Al : l \in L\} \quad (6.1)$$

Повторюючи цей процес, утворюємо послідовність підпросторів Крилова

$$K(L) = \{L, AL, A^2L, \dots\} \quad (6.2)$$

На практиці L задається опосередковано через ортонормований базис, скажімо $S = (x_1, x_2, x_3)$. Тоді $A^k L$ буде натягнуто на $A^k S = (A^k x_1, A^k x_2, A^k x_3)$. Навіть якщо ці стовбці нормовані, вони просто є k -ми членами в трьох окремих послідовностях, кожна з яких буде сходитися (повільно) до домінуючого власного вектору z_n , де $Az_n = z_n \lambda_n$ та $\lambda_n = \|A\|$. Три послідовності, що повільно сходяться, можуть бути кращими за одну, тому що $A^k S$ — поганий базис для гарного підпростору $A^k L$.

Якщо система виду $(A - \sigma)x = b$ легко піддається розв'язанню або шляхом факторизації матриці $A - \sigma$, або ітераціями, тоді метод ітерування підпростору з матрицею $(A - \sigma)^{-1}$ можна ефективно використати для отримання p найближчих до σ власних значень разом з відповідними їм власними векторами. Це найбільш розповсюдженний спосіб використання цієї методики.

Основним моментом тут є те, що винагородою за одночасну роботу з декількома стовпцями з періодичним ортонормуванням є покращений коефіцієнт лінійної збіжності для послідовних підпросторів. Коли потрібно знайти декілька дуже близьких один до одного власних значень, покращення особливо помітно і це повністю компенсує додаткову роботу, спричинену використанням підпросторів більшої розмірності, ніж це потрібно насправді. В інших випадках метод може виявитися дуже повільним. Труднощі поля-

гають в тому, що розподілення власних значень зазвичай невідомо, і тому незрозуміло, наскільки великим має бути обрано p -розмірність підпростору L . Більш того, сама ефективність методу сильно залежить від значення величини p .

Протягом 1960-1970 років, коли алгоритм Ланцоша перебував у забутті, були розроблені тонкі версії методу ітерування підпросторів. Але не зважаючи на те, що зараз, коли існують прості у використанні і надійні програми, що реалізують алгоритм Ланцоша, не слід відмовлятися від методу ітерування підпростору. Існують декілька ситуацій, в яких його застосування виправдано.

1. Якщо в наявності немає вторинної пам'яті, а в оперативній пам'яті може одночасно зберігатися лише кілька n -мірних векторів, немає іншого виходу, як відкидати попередні вектори послідовності Крилова і використовувати ітерування підпростору.

2. Якщо відносна відстань між шуканими власними значеннями та рештою дуже велика, як в зворотніх ітераціях зі зсувом, то для збіжності знадобиться тільки один-два степеневі кроки методу. Це настільки гарна ситуація, що перевага методу Ланцоша тут зовсім не потрібна. Тим не менш за таких сприятливих обставин метод Ланцоша також дуже добре працює.

Найкращим прикладом того, наскільки добре метод ітерування підпростору може бути пристосований до автоматизації обчислень, є програма Рутісхаузера RITZIT. Програма достатньо складна, щоб бути ефективною для широкого кола застосувань. Тим не менш вона зрозуміла, і при її створенні було витрачено значних зусиль, щоб скоротити кількість питань, з яких користувач повинен приймати самостійні рішення.

Реалізації

Опис методу буде спрощено припущенням, що A додатно визначена і що треба знайти p домінуючих власних пар (α_i, z_i) , $i = n - p + 1$. Матриця не буде змінюватися і не може бути відома в явному вигляді, оскільки її роль в програмі відповідає оператору, що отримує в якості аргументу вектори u і що повертає (в якості результату) вектори Au . Будь які спеціальні особли-

вості, які дозволяють економити пам'ять або арифметичні операції, можуть бути враховані користувачем під час написання підпрограми множення матриці на вектор, яку Рутісхаузер називає ОР (від англійського слова "operator— оператор).

Для кожної реалізації v -ий крок перетворює ортонормальний базис S_{v-1} підпростору $A^{v-1}L$ в ортонормальний базис S_v підпростору A^vL . Треба додати, що має бути певний критерій збіжності. (Прямокутна матриця S_v не пов'язана жодним чином з матрицею S_j , складеної із власних векторів матриці T_j)

Алгоритм 6.1. Реалізація 1. Просте ітерування підпростору

- 1) Обчислюємо $C_v = AS_{v-1}$
- 2) Перевіряємо критерій збіжності для кожного стовбця
- 3) Ортонормуємо $C_v = Q_v R_v$ (модифікованим процесом Грама-Шмідта)
- 4) Призначаємо $S_v = Q_v$

Зауваження 6.1. Стовбці S_v не є оптимальними апроксимаціями з $\text{span } S_v$ до шуканих власних векторів. Навіть якщо $L = \text{span}(z_n, z_{n-1})$, то стовбці S_v будуть збігатися при $v \rightarrow \infty$ тільки лінійно до z_n і z_{n-1} , не дивлячись на те, що вони вже знаходяться в $L = \text{span } S_0$. Цей недолік наводить на думку, що процедура Релея-Рітца повинна часто застосовуватися до підпростору S_v .

Алгоритм 6.2. Реалізація 2. Об'єднання ітерування підпростору та процедури Релея-Рітца

- 1) Обчислюємо $C_v = AS_{v-1}$
- 2) Ортонормуємо $C_v = Q_v R_v$ (модифікованим процесом Грама-Шмідта)
- 3) Формуємо $\hat{H}_v = Q_v^*(A Q_v)$
- 4) Розкладаємо $\hat{H}_v = G_v \Theta_v G_v^*$
- 5) Формуємо $S_v = Q_v G_v$ векторів Рітца для $A^v L$

Зауваження 6.2. Якщо G_v знайдено як добуток ортонормальних матриць, скажімо $G_v = P_1 \dots P_k$, то 5) може бути виконано одночасно з 4) за наступним алгоритмом: $S_v = Q_v$ і далі $S_v \leftarrow S_v P_i$ для $i = 1, \dots, k$. Формувати

матрицю G_v у явному вигляді немає необхідності, а для зберігання S , C і Q достатньо одного масива розмірності $n \times p$.

S_v є найкращим базисом в $A^v L$, а її стовбці сходяться до векторів z . Проте витрати при цьому значні, зокрема операція 3) потребує p додаткових викликів підпрограми ОР, що призводить до подвоєння вартості кожного кроку в порівнянні з реалізацією 1. На щастя, є розумний спосіб уникнути цих додаткових викликів, заснований на факті, що $\phi(A)$ має ті ж самі власні вектори, що і матриця A , для будь-якої аналітичної функції ϕ , значення якої для різних власних значень (як значеннях аргумента) також різні.

Поки залишимо функцію ϕ невизначеною і знайдемо апроксимації Релея-Рітца до $\phi(A)$ з $A^v L$. Аналогічно попереднім реалізаціям припустимо

$$C_v = AS_{v-1}. \quad (6.3)$$

Знайдемо новий базис, наприклад $C_v F_v$, $s(p \times p)$ -матрицею F_v , що задовільняє двом умовам. А саме, умові ортонормальності

$$(C_v F_v)^*(C_v F_v) = 1_p \quad (6.4)$$

та умові

$$(C_v F_v)^* \phi(A) (C_v F_v) = \Delta_v^{-2}, \quad (6.5)$$

(що приводить до векторів Рітца), де Δ_v^{-2} — діагональна матриця. Правильним вибором є $\phi(\xi) = \xi^{-2}$, так що (6.5) перетворюється в співвідношення

$$(F_v^* S_{v-1}^* A) A^{-2} (AS_{v-1} F_v) = F_v^* F_v = \Delta_v^{-2}. \quad (6.6)$$

Звідси випливає, що $F_v \Delta_v$ буде ортонормальною матрицею і (6.4) приймає, відповідно (6.6), наступний вигляд:

$$C_v^* C_v = (F_v^{-*} \Delta_v^{-1}) \Delta_v^2 (\Delta_v^{-1} F_v^{-1}) = (F_v \Delta_v) \Delta_v^2 (F_v \Delta_v)^*. \quad (6.7)$$

Таким чином, F_v і Δ_v визначаються шляхом спектрального розкладу матриці $C_v^* C_v$.

Алгоритм 6.3. Реалізація 3. Це реалізація 2, застосована до A^{-2}

- 1) Обчислюємо $C_v = AS_{v-1}$
- 2) Обчислюємо $\mathring{H}_v = C_v^* C_v$
- 3) Виконуємо спектральний розклад $\mathring{H}_v = B_v \Delta_v^2 B_v^*$
- 4) Формуємо $S_v = C_v B_v \Delta_v^{-1} (= C_v F_v)$

Зауваження 6.3. $\mathring{H}_v$ є проєкцією A^2 на $\text{span } S_{v-1}$. Нове S_v буде відрізнятися від S_v з попередньої реалізації. Крім того, S_v не представлено у вигляді добутку ортонормальних матриць. Райнш знайшов розумний спосіб усунути цей пробіл наступним чином. Нехай QR -розклад матриці C_v має вигляд $Q_v R_v$, як і в реалізації 2. Тоді з операції 4) реалізації 3 маємо

$$\mathring{H}_v = C_v^* C_v = R_v^* Q_v^* Q_v R_v = R_v^* R_v. \quad (6.8)$$

LR -перетворення було попередником QR -перетворення, і, будучи застосованим, до $\mathring{H}_v$, воно приводить до матриці H_v (більш близької до діагональної), що визначається перестановкою співмножників матриці

$$\begin{aligned} H_v &\equiv R_v R_v^* = R_v \mathring{H}_v R_v^{-1} = R_v B_v \Delta_v^2 B_v^* R_v^{-1} = \\ &= (R_v B_v \Delta_v^{-1}) \Delta_v^2 (\Delta_v B_v^* R_v^{-1}) \end{aligned} \quad (6.9)$$

(тут була використана операція 3) з реалізації 3). Можна перевірити, що $P_v = R_v B_v \Delta_v^{-1}$ є ортонормальною матрицею, складеною з власних векторів H_v . Тому з операції 3) реалізації 3 маємо

$$S_v = C_v B_v \Delta_v^{-1} = Q_v R_v B_v \Delta_v^{-1} = Q_v P_v. \quad (6.10)$$

Алгоритм 6.4. Реалізація 4 (за програмою RITZIT)

- 1) Обчислюємо $C_v = AS_{v-1}$
- 2) Розкладаємо $C_v = Q_v R_v$
- 3) Формуємо $H_v = R_v R_v^*$
- 4) Розкладаємо $H_v = P_v \Delta_v^2 P_v^*$
- 5) Формуємо $S_v = Q_v P_v$

Зауваження 6.4. Операції 4) та 5) можуть бути здійснені разом.

Удосконалення

Чебишове прискорення

Ця методика застосовується в багатьох областях чисельного аналізу і, тому заслуговує уваги. Навіть у зв'язку з реалізацією 4 процедура Релея-Рітца залишається дорогою, і тому заманливо зробити декілька кроків основного степеневому методу між кожним застосуванням процедури Релея-Рітца. Для визначеності припустимо, що алгоритм обчислює послідовність $S_{v+i} = AS_{v+j-1}$, $j = 1, \dots, m$. Проте замість цього не складніше обчислювати послідовність $S_{v+j} = (A - \sigma_j)S_{v+j-1}$, $j = 1, \dots, m$, ввівши зсуви σ_j . Таким чином алгоритм міг би обчислювати послідовність

$$S_{v+m} = \phi(A)S_v \quad (6.11)$$

з будь-яким нормованим поліномом ступеня m . Проблема полягає в тому, щоб обрати поліном ϕ .

Після першого застосування процедури Релея-Рітца, програма володіє апроксимаціями $\theta_{-1}, \dots, \theta_{-p}$ до домінуючих власних значень. θ_{-i} є власними значеннями H_v , або $\dot{H}_v$, або $\hat{H}_v$. Поміркувавши стає зрозуміло, що корисним (це не означає, що оптимальним) є вибір такого ϕ , який на інтервалі $[\theta_1, \theta_{-p}]$ настільки малий, наскільки це можливо. Оскільки θ_1 невідомо, то зазвичай використовують інтервали $[0, \theta_{-p}]$, якщо матриця A додатно визначена, і $[-\theta_{-p}, \theta_{-p}]$ в іншому випадку. Сформульована задача в теорії апроксимації розв'язується за допомогою поліномів Чебишова, адаптованих до відповідного інтервалу. Чудовим фактом є те, що тут не потрібно знати, а після використовувати нулі відповідних поліномів Чебишова $T_m(\xi)$ в якості зсувів σ_j , оскільки проста тричленна рекурсія з постійними коефіцієнтами дозволяє обчислювати $T_j(A)$ через $T_{j-1}(A)$ і $T_{j-2}(A)$. Точніше кажучи, якщо T_m адаптований до інтервалу $[-\theta, \theta]$, то

$$\left\{ \begin{array}{l} T_{v+j}(\xi) = \frac{2\xi}{e}T_{v+j-1}(\xi) - T_{v+j-2}(\xi), \\ S_{v+j} = \frac{2}{e}AS_{v+j-1} - S_{v+j-2}, \end{array} \right\} \quad j = 2, \dots, m. \quad (6.12)$$

Крім того, зберігати S_{v+j-2} немає необхідності при умові, що всі стовбці S_v по черзі перетворюються безпосередньо до стовбців S_{v+m} . Це означає, що додатково потрібно зберігати тільки два n -мірних вектори, щоб реалізувати пропонуване прискорення збіжності S_v .

Щодо вибору m , важлива перевага рекурсії полягає у тому, що вона не залежить від m і, відтак, значення m може бути обрано в процесі обчислень. Множник збіжності для домінуючого вектора Рітца дорівнює $\theta_{-p-1}/\theta_{-1}$. Якщо це відношення мале, наприклад 0.1, то процес майже не потребує прискорення, значення m може бути обрано рівним одиниці. Проте, коли відношення близьке до одиниці, скажімо 0.98, то великі значення m дають помітну користь. Єдиним обмеженням є вимога, щоб стовбці матриць S_{v+m} були повністю незалежними. В своїй програмі RITZIT Рутісхаузер вимагає, щоб

$$\|S_{v+m}\| \doteq T_m(\theta_{-1}/\theta_{-p}) \leq \text{ch8} < 1500. \quad (6.13)$$

Згадана програма, що включає в себе цей потужний механізм, виключно проста і дуже елегантна. Тим не менш, щоб притримуватися правильної перспективи, треба нагадати, що алгоритм Ланцоша через m кроків дає більше, ніж чебишове прискорення, що використовує невідомий оптимальний інтервал α_1, α_{-p-1} .

Рандомізація

Є спосіб захистити алгоритм від невдалого вибору початкового підпростору L , яке може виявитися майже ортогональним до одного з шуканих власних векторів. Після кожної ортогоналізації базисних векторів, один з них з найвнутрішнім відношенням Релея θ_{-p} замінюється випадковим вектором, ортогональним до решти векторів базису. Це найпростіший спосіб зробити якнайменш ймовірний пропуск будь-якого шуканого власного вектора.

Очевидно варто чекати, поки не стабілізується відношення Релея, і Рутісхаузер вимагає виконання трьох апроксимацій Релея-Рітца. Це приклад одного з рідкісних спеціальних параметрів, що є в програмі. Альтернатива, що полягає в обчисленні відношення Релея до стабілізації

θ_{-m} видається надмірно складною.

Зауважимо, що цей механізм перешкоджає чебишовому прискоренню, яке використовує величину p -го відношення Релея θ_{-p} .

Апроксимація Релея-Рітца θ_i без округлень монотонно наближається до своїх границь α_i ($i = \pm 1, \pm 2, \dots$) при продовженні ітерування підпростору.

Збіжність

Нехай $Z \equiv (z_1, \dots, z_m)$, де $Az_i = z_i\alpha_i$, $i = 1, \dots, m$ — матриця шуканих власних векторів. Для визначеності ми припустимо

$$0 < \alpha_1 \leq \alpha_2 \leq \dots \leq \alpha_m < \alpha_{m+1} \leq \dots, \quad (6.14)$$

так, щоб $W = W^m = \text{span } Z$ було домінуючим інваріантним підпростором матриці A^{-1} . Нехай L — довільне m -мірний підпростір E^n і $\{A^{-k}L : k = 0, 1, 2, \dots\}$ — відповідна послідовність, утворена ітеруванням підпростору L . Величина $\angle(W, A^{-k}L)$ є занадто грубою мірою якості апроксимації W підпростору $A^{-k}L$, оскільки цікаво знати, наскільки добре окремі вектори, такі як z_1 , може бути апроксимовано векторами з $A^{-k}L$. Тому об'єктами дослідження є величини

$$\psi_j^{(k)} \equiv \angle(z_j, A^{-k}L) \equiv \min \angle(z_j, x) \quad \text{по всім } x \in A^{-k}L. \quad (6.15)$$

Потрібно знати як швидко $\psi_i^{(k)} \rightarrow 0$, в той же час хотілося б довести настільки близьким до одновимірного випадку, наскільки можливо.

Щоб при $k \rightarrow \infty$ послідовність $\{A^{-k}L\}$ збігалася до W , а не до іншого інваріантного підпростору, необхідно припустити чи виконання нерівності $\angle(W, L) < \frac{\pi}{2}$, чи, еквівалентно, що для будь якого ортонормованого базису S підпростору L матриця

$$Z * S \quad \text{обернена.} \quad (6.16)$$

Корисне поняття в аналізі — це кут Ψ (матричний) між W та L , що

визначається співвідношенням

$$\Psi = \arccos(Z^* S S^* Z)^{\frac{1}{2}}. \quad (6.17)$$

Функції від матриць можуть бути визначені різними способами, але у нашому випадку матричний аргумент Ψ необхідний тільки для того, щоб надати сенс матрицям, як $\sin \Psi$ і $\tan \Psi$, які, будучи вираженими у термінах S та Z , мають не дуже зручний вигляд. Припущення (6.16) гарантує, що Ψ визначена коректно. Зауважимо, що вона не є діагональною.

За аналогією з одномірним випадком існує такий ортогональний базис S , що $Z^* S = S^* Z = \cos \Psi$. Цей базис S може бути виражено через Z наступним чином:

$$S = Z \cos \Psi + J \sin \Psi, \quad (6.18)$$

де

$$J^* J = I_m, \quad Z^* J = O. \quad (6.19)$$

Теорема 6.1. *При зроблених щодо L і W припущеннях (6.14) і (6.16) кожний власний вектор z_i , $i \leq m$, задовільняє нерівності*

$$\tan \angle(z_i, A^{-k} L) \leq \left(\frac{\alpha_i}{\alpha_{m+1}} \right)^k \tan \angle(W, L). \quad (6.20)$$

Доведення. Помноживши рівність (6.18) зліва на матрицю A^{-k} , а після справа на матрицю $(\sec \Psi) \Lambda^k$, отримаємо

$$\begin{aligned} A^{-k} S (\sec \Psi) \Lambda^k &= A^{-k} Z \Lambda^k + A^{-k} J (\tan \Psi) \Lambda^k \\ &= (Z \Lambda^{-k}) \Lambda^k + A^{-k} J (\tan \Psi) \Lambda^k, \end{aligned} \quad (6.21)$$

де $\Lambda = \text{diag}(\alpha_1, \dots, \alpha_m)$. Ключовим фактом є ортогональність Z до $A^{-k} J$. Тому зручно переписати $A^{-k} J$ у вигляді

$$A^{-k} J = J_k \Omega_k, \quad \Omega_k = (J^* A^{2k} J)^{\frac{1}{2}}, \quad (6.22)$$

так щоб матриця J_k була ортонормальною. Зауважимо, що оскільки $Z^* J =$

О, то

$$\|\Omega_k\|^2 \max_{\|v\|=1} v^* J^* A^{-2k} J v \leq \alpha_{m+1}^{-2k}. \quad (6.23)$$

Розглянемо тепер j -й стовбець з (6.21):

$$x_j^{(k)} \equiv A^{-k} S(\sec \Psi) e_j \alpha_j^k = z_j + u_j, \quad (6.24)$$

де $u_j = J_k \Omega_k(\tan \Psi) e_j \alpha_j^k$. Звідси, послідовно використовуючи (6.22) та (6.23), остаточно отримуємо

$$\begin{aligned} \tan \angle(z_j, A^{-k} L) &\leq \tan \angle(z_j, x_j^{(k)}) = \|u_j\|/1 = \\ &= \|\Omega_k(\tan \Psi) e_j\| \alpha_j^k \leq \\ &\leq \alpha_j^k \|\Omega_k\| \|(\tan \Psi) e_j\| \leq \\ &\leq \left(\frac{\alpha_j}{\alpha_{m+1}} \right)^k \tan \angle(z_j, S(\sec \Psi) e_j) \leq \\ &\leq \left(\frac{\alpha_j}{\alpha_{m+1}} \right)^k \tan \angle(W, L). \end{aligned} \quad (6.25)$$

Теорема доказана. □

Оцінити цей результат можна, зіставивши обчислення z_1 шляхом ітерування підпростору і методом зворотньої ітерації.

Теорема 6.1 показує, що деяка послідовність $\{x_j^{(k)} : x_j^{(k)} = A^{-k} S(\sec \Psi) e_j\} \rightarrow z_j$ при $k \rightarrow \infty$, але це не відноситься до поведінки послідовності $\{y_j^{(k)}\}$, обчисленої в дійсності, скажімо, за допомогою використання реалізації 4.

Рутісхаузер показує, що $\{y_j^{(k)}\} \rightarrow \{x_j^{(k)}\}$ з тою ж самою асимптотичною швидкістю, що і $\{x_j^{(k)}\} \rightarrow \{z_j^{(k)}\}$, тоді лінійну збіжність $\{y_j^{(k)}\}$ до z_j визначає той же самий покращений множник $\left(\frac{\alpha_j}{\alpha_{m+1}} \right)$. Складність доведення пояснюється складною природою векторів Рітца, і необхіден короткий відступ, перш ніж ці твердження можна буде обґрунтувати. Якщо B є неортонормальним базисом для $\text{span } B$, то кожна апроксимація Рітца для A^2 з $\text{span } B$ має вигляд Bt , де t задовільняє рівнянню

$$(B^* A^2 B - \mu^2 B^* B) t = o \quad (6.26)$$

при відповідному значенні Рітца μ .

Теорема 6.2. *Якщо метод ітерування підпростору використовує реалізацію 4, то кожен вектор Рітца $y_i^{(k)}$ при $k \rightarrow \infty$ пов'язаний з вектором $x_i^{(k)}$ теореми 6.1 відношенням*

$$\sin \angle(y_i^{(k)}, x_i^{(k)}) = O \left[\left(\frac{\alpha_j}{\alpha_{m+1}} \right)^k \right], \quad i = 1, \dots, m. \quad (6.27)$$

Доведення. В базисі $B = A^{-k} S(\sec \Psi) \Lambda^k$, що задано в (6.21), $x_i^{(k)}$ представлено вектором e_i , а $y_i^{(k)}$ представлено розв'язком t рівняння (6.26). Після підстановки правої частини (6.21) замість B рівняння (6.26) приймає вигляд

$$[\Lambda^2 - \mu^2 + (\alpha_{m+1}^{-1} \Lambda)^k H_k (\alpha - m + 1^{-1} \Lambda)^k] t = o, \quad (6.28)$$

де

$$\|H_k\| \leq (\alpha - m + 1 \|\tan \Psi\|)^2. \quad (6.29)$$

Збурення в (6.28) до $\Lambda^2 - \mu^2$ при $k \rightarrow \infty$ перетворюється на нуль, тому для достатньо великих k існують μ_i , близькі до α_i і t_i , близькі до e_i . Розглянемо тепер k , настільки велике, що

$$|\mu_i - \alpha_i| \leq \delta \equiv \frac{1}{2} \min |\alpha_j - \alpha_i| \quad \text{по всім} \quad \alpha_j \neq \alpha_i. \quad (6.30)$$

При таких k (α_i^2, t_i) є гарною апроксимуючою парою для власної пари (μ_i^2, e_i) , і вектор нев'язки r_i для (6.28) задовільняє відношенням

$$\begin{aligned} \|r_i\| &= \|[\Lambda^2 + (\alpha_{m+1}^{-1} \Lambda)^k H_k (\alpha_{m+1}^{-1} \Lambda)^k - \alpha_i^2] e_i\| = \\ &= \|(\alpha_{m+1}^{-1} \Lambda H_k e_i\| \left(\frac{\alpha_i}{\alpha_{m+1}} \right)^k \leq \\ &\leq \left(\frac{\alpha_m}{\alpha_{m+1}} \right)^k \left(\frac{\alpha_i}{\alpha_{m+1}} \right)^k (\alpha_{m+1} \|\tan \Psi\|)^2. \end{aligned} \quad (6.31)$$

При виведенні останньої нерівності використовується (6.29). За теоремою про результати про відокремлення

$$\sin \angle(t_i, e_i) \leq \frac{\|r_i\|}{\delta}. \quad (6.32)$$

Для великих k базис B майже ортонормальний, оскільки

$$|b_i^* b_j| \leq \left(\frac{\alpha_i}{\alpha_{m+1}} \right)^k \left(\frac{\alpha_j}{\alpha_{m+1}} \right)^k \|\tan \Psi\|^2, \quad i \neq j. \quad (6.33)$$

Якщо об'єднати нерівності (6.32) і (6.33), то виявляється, що в границі (6.33) домінує і при $k \rightarrow \infty$

$$\sin \angle(y_i^{(k)}, x_i^{(k)}) \leq \left(\frac{\alpha_i}{\alpha_{m+1}} \right)^k \|\tan \Psi\|. \quad (6.34)$$

Теорема доказана. □

РОЗДІЛ 7

ОБЧИСЛЮВАЛЬНИЙ ЕКСПЕРИМЕНТ

Задача скінченно-різницевої апроксимації спектральної задачі для оператора Лапласа в одиничному прямокутнику з однорідними умовами Діріхле полягає у знаходженні власних значень для симетричної матриці A . Власні значення матриці A є відмінними. Мінімальне власне значення та відповідний йому власний вектор визначаються так:

$$\lambda_h = \frac{8}{h^2} \sin^2 \frac{\pi h}{2},$$

$$\varphi_h(x_i, y_i) = 2 \sin(\pi x_i) \sin(\pi y_i), \quad x_i = ih, y_j = jh, i, j = 1, \dots, N-1.$$

Тут число N (ціле) визначає параметр $h = 1/N$ для рівномірної сітки на одиничному прямокутнику.

Розрахунки виконувалися у середовищі MATLAB. Методом Ньютона та ітерації зворотної степені з використанням методу Релея визначалися мінімальне власне значення та відповідний власний вектор матриці A розмірності $10^5 \times 10^5$ ($N = 101$). Щоб визначити початкові наближення λ^0 і x^0 виконувався один етап ітерації зворотної степені, використовуючи початковий вектор $y^0 = [1, \dots, 1]^T$.

Результати обчислень можна побачити у таблицях 1 та 2. Важливо відзначити, що після трьох ітерацій методу зворотної степені за використанням методу Релея матриця $A - \rho^h I$ стає виродженою у машинній арифметиці, що призводить до припинення обчислень. Натомість, у методі Ньютона число обумовленості матриці системи залишається незмінним під час наближення до власного значення, дозволяючи продовжувати обчислення для досягнення більш високої точності.

Табл. 7.1. Зворотна ітерація з відношенням Релея.

k	$\ r^k\ _2$	$\lambda_h - \rho^k$	$\ \varphi_h - x^k\ _2$	$\text{cond}(A - \rho^k I)$
1	12.2	-0.901	2.00	9.04e+04
2	0.0895	-9.64e-05	1.33e-09	8.46e+08
3	1.06e-07	-1.42e-14	*	6.08e+16

Табл. 7.2. Метод Ньютона.

k	$\ r^k\ _2$	$\lambda_h - \lambda^k$	$\ \varphi_h - x^k\ _2$	$\text{cond} \left(\begin{bmatrix} A - \lambda^k I & -x^k \\ -(x^k)^T & 0 \end{bmatrix} \right)$
1	14.2	0.0874	0.0125	6.83e+05
2	0.900	5.04e-04	7.801e-05	8.47e+04
3	0.00108	3.88e-08	3.05e-09	8.16e+04
4	3.93e-08	-3.55e-14	1.84e-15	8.16e+04
5	4.25e-12	-7.11e-15	1.77e-15	8.16e+04

ВИСНОВКИ

Встановлено і доведено збіжність методу Ньютона в околі простого власного значення симетричної матриці. А також запропоновано використовувати метод Ньютона для знаходження власних значень і векторів симетричної матриці.

Якщо порівнювати метод Ньютона та метод зворотної ітерації з використанням відношення Релея, то вони вимагають практично однакову кількість ітерацій, але обчислювальні витрати майже ідентичні. Хоча зворотна ітерація збігається швидше, метод Ньютона дозволяє отримати власне значення з більшою точністю.

СПИСОК ЛІТЕРАТУРИ

1. Гаврилюк І.П., Макаров В.Л. Методи обчислень. Підручник. У 2ч. / І.П. Гаврилюк, В.Л. Макаров — К.: Вища шк., 1995. — 367 с.
2. Шахно С. Чисельні методи лінійної алгебри / С. Шахно — Львів: Видавничий центр ЛНУ ім. І. Франка, 2007. — 245 с .
3. Bai Z., Demmel J., Dongarra J., Ruhe A., and H. van der Vorst. Templates for the Solution of Algebraic Eigenvalue Problems: A Practical Guide. / Z. Bai, J. Demmel, J. Dongarra, A. Ruhe, та H. van der Vorst — SIAM, Philadelphia, PA, USA, 2000. — 410 p.
4. Beilina L., Karchevskii E., Karchevskii M. Numerical linear algebra: theory and applications / ChatGPT L. Beilina, E. Karchevskii, M. Karchevskii. — Springer, 2017. 457 p.
5. Demmel James W. Applied numerical linear algebra. / James W. Demmel — SIAM, 1997. — 425 p.
6. Epperson J. F. An Introduction to Numerical Methods and Analysis : Third edition / J. F. Epperson — John Wiley & Sons Inc., 2021. 286 p.
7. Gupta R. K. Numerical methods. Fundamentals and applications / R. K. Gupta — Cambridge University Press, 2019. 830 p.
8. Higham Nicholas J., Functions of Matrices: Theory and Computation / Nicholas J. Higham — SIAM, 2008. — 425 p.
9. Vandebril R., David S., Aurentz W. J. L., Mach T., Robol L. Core-Chasing Algorithms for the Eigenvalue Problem / R. Vandebril, S. David, W. J. L. Aurentz, T. Mach, L. Robol — SIAM, 2018.
10. Lyche T. Numerical linear algebra and matrix factorizations / T. Lyche — Springer, 2020. — 376 p.
11. Saad Y. Iterative Methods for Sparse Linear Systems, Second Edition / Y. Saad — SIAM, 2003. — 350 p.
12. Saad Y. Numerical Methods for Large Eigenvalue Problems - 2nd Edition / Y. Saad — SIAM, 2011, 285 p.
13. Vorst H. A. Iterative Krylov Methods for Large Linear systems / H. A. Vorst. — Cambridge University Press, Cambridge, 2003. — 221 p.

14. Verbitskyi V. V., Huk A. G. Newton's method for the eigenvalue problem of a symmetric matrix / V. V. Verbitskyi — Odesa: Odesa I. I. Mechnikov National Univ. scientific journal, 2020. — 75 c.
15. Wilkinson J. H. The algebraic eigenvalue problem. / J. H. Wilkinson — New York: Oxford Univ. press, 1965.
16. Zhang T., Golub G. H., Law K.H. Subspace iterative methods for eigenvalue problems / T. Zhang, G. H. Golub, K.H Law. — New York: Elsevier, 1999.
17. Wang B., Xie H. A New Subspace Iteration Algorithm for Solving Generalized Eigenvalue Problems / B. Wang, H. Xie — ResearchGate, 2022.
18. Arnoldi W. E. The principle of minimized iterations in the solution of the matrix eigenvalue problem. Quarterly of Applied Mathematics / W. E. Arnoldi 1951.
19. Balay, S., Abhyankar, S., Adams, M., Brown, J., Brune, P., Buschelman, K., Dalcin, L., Dener, A., Eijkhout, V., Gropp, W., et al. PETSc users manual / S. Balay, S. Abhyankar, M. Adams, J. Brown, P. Brune, K. Buschelman, L. Dalcin, A. Dener, V. Eijkhout, W. Gropp, et al. — 2019.
20. Bradbury, W., Fletcher, R.: New iterative methods for solution of the eigenproblem. Numerische Mathematik 9(3) / W. Bradbury, R. Fletcher. — 1966. — 259–267 p.
21. Davidson, E. The iterative calculation of a few of the lowest eigenvalues and corresponding eigenvectors of large real-symmetric matrices / E. Davidson. — Journal of Computational Physics 17, 1975 — 87–94 p.
22. Fokkema, D.R., Sleijpen, G.L., Van der Vorst, H.A. Jacobi–Davidson style QR and QZ algorithms for the reduction of matrix pencils / D.R. Fokkema, G.L. Sleijpen, H.A. Van der Vorst. — SIAM Journal on Scientific Computing 20(1), 1998. — 94–125 p.
23. Golub, G.H., Van der Vorst, H.A. Eigenvalue computation in the 20th century / G.H. Golub, H.A. Van der Vorst — Journal of Computational and Applied Mathematics, 2000. — 35–65 p.
24. Hecht, F. New development in FreeFem++ / F. Hecht — Journal of Numerical Mathematics 20(3-4), 2012. — 251–266 p.
25. Hestenes, M.R., Karush, W. A method of gradients for the calculation of the characteristic roots and vectors of a real symmetric matrix / M.R. Hestenes,

- W. Karush — Journal of Research of the National Bureau of Standards 47(1), 1951. — 45–61 p.
26. Hochstenbach, M.E., Sleijpen, G.L. Two-sided and alternating Jacobi–Davidson / M.E. Hochstenbach, G.L. Sleijpen — Linear Algebra and its Applications 358(1-3), 2003. — 145–172 p.
 27. Jia, Z. On convergence of the inexact Rayleigh quotient iteration with MINRES / Z. Jia. — Journal of Computational and Applied Mathematics 236(17), 2012. — 4276–4295 p.
 28. Knyazev A.V. Toward the optimal preconditioned eigensolver: locally optimal block preconditioned conjugate gradient method / A.V. Knyazev. — SIAM Journal on Scientific Computing 23(2), 2001. — 517–541 p.
 29. Lanczos C. An iteration method for the solution of the eigenvalue problem of linear differential and integral operators / C. Lanczos. — 1950.
 30. Lehoucq, R.B., Meerbergen, K. Using generalized Cayley transformations within an inexact rational Krylov sequence method / R.B. Lehoucq, K. Meerbergen. — SIAM Journal on Matrix Analysis and Applications 20(1), 1998. — 131–148 p.
 31. Li, Y., Wang, Z., Xie, H. GCGE: A package for solving large scale eigenvalue problems by parallel block damping inverse power method / Y. Li, Z. Wang, H. Xie. — 2021.
 32. Miao, C.Q. On Chebyshev–Davidson method for symmetric generalized eigenvalue problems / C.Q. Miao. — Journal of Scientific Computing 85(3), 2020. — 1–22 p.
 33. Morgan, R.B., Scott, D.S. Generalizations of Davidson’s method for computing eigenvalues of sparse symmetric matrices / R.B. Morgan, D.S. Scott. — SIAM Journal on Scientific and Statistical Computing 7(3), 1986. — 817–825 p.
 34. Parlett, B.N.: The Symmetric Eigenvalue Problem / B.N. Parlett — SIAM, 1998.
 35. Roman, J.E., Campos, C., Romero, E., Tomas, A. SLEPc users manual. D. Sistemes Informatics i Computacio Universitat Politecnica de Valencia, Valencia, Spain, Report No. DSIC-II/24/02, 2015.
 36. Ruhe, A.: Rational Krylov sequence methods for eigenvalue computation /

- A. Ruhe. — Linear Algebra and its Applications 58, 1984. — 391–405 p.
37. Saad, Y. Numerical Methods for Large Eigenvalue Problems: revised edition / Y. Saad. — SIAM, 2011.
 38. Simoncini, V., Elden, L. Inexact Rayleigh quotient-type methods for eigenvalue computations / V. Simoncini, L. Elden. — BIT Numerical Mathematics 42(1), 2002. — 159–182 p.
 39. Sleijpen, G.L., Booten, A.G., Fokkema, D.R., Van der Vorst, H.A. Jacobi–Davidson type methods for generalized eigenproblems and polynomial eigenproblems / G.L. Sleijpen, A.G. Booten, D.R. Fokkema, H.A. Van der Vorst — BIT Numerical Mathematics 36(3), 1996. — 595–633 p.
 40. Sleijpen, G.L., Van der Vorst, H.A. A Jacobi–Davidson iteration method for linear eigenvalue problems / G.L. Sleijpen, H.A. Van der Vorst. — SIAM Review 42(2), 2000. — 267–293 p.
 41. Sorensen, D.C. Implicit application of polynomial filters in a k-step Arnoldi method / D.C. Sorensen — SIAM Journal on Matrix Analysis and Applications 13(1), 1992. — 357–385 p.
 42. Sorensen, D.C. Numerical methods for large eigenvalue problems / D.C. Sorensen — Acta Numerica 11, 2002. — 519–584 p.
 43. Stewart, G.W. A Krylov–Schur algorithm for large eigenproblems / G.W. Stewart — SIAM Journal on Matrix Analysis and Applications 23(3), 2002. — 601–614 p.
 44. Szyld, D.B., Xue, F. Efficient preconditioned inner solves for inexact Rayleigh quotient iteration and their connections to the single-vector Jacobi–Davidson method / D.B. Szyld, F. Xue — SIAM Journal on Matrix Analysis and Applications 32(3), 2011. — 993–1018 p.
 45. Vecharynski, E., Yang, C., Pask, J.E. A projected preconditioned conjugate gradient algorithm for computing many extreme eigenpairs of a Hermitian matrix / E. Vecharynski, C. Yang, J.E. Pask — Journal of Computational Physics 290, 2015. — 73–89 p.
 46. Van der Vorst, H.A. Computational methods for large eigenvalue problems / H.A. Van der Vorst — 2002.
 47. Zhou, Y. Studies on Jacobi–Davidson, Rayleigh quotient iteration, inverse iteration generalized Davidson and Newton updates / Y. Zhou — Numerical

- Linear Algebra with Applications 13(8), 2006. — 621–642 p.
48. Zhou, Y., Saad, Y. A Chebyshev–Davidson algorithm for large symmetric eigenproblems / Y. Zhou, Y. Saad — SIAM Journal on Matrix Analysis and Applications 29(3), 2007. — 954–971 p.

ДОДАТОК А. КОД ПРОГРАМИ

```

function NewtonRaphsonIteration4

    gridSize = 40;
    laplaceMatrix = initializeLaplaceMatrix(gridSize);

    dimension = size(laplaceMatrix, 1);
    smallestEigenvalue = eigs(laplaceMatrix, 1, 'SM');
    lambda1_h = 8 * gridSize * gridSize *
        sin(pi / 2 / gridSize) * sin(pi / 2 / gridSize);
    randomVector = rand(dimension, 1);
    initialVector = randomVector / norm(randomVector, 2);

    for iteration = 1:3
        updatedVector = laplaceMatrix \ initialVector;
        currentEigenvalue = 1 / (updatedVector' * initialVector);
        initialVector = updatedVector / norm(updatedVector, 2);
    endfor

    originalVector = initialVector;
    originalEigenvalue = currentEigenvalue;
    printf('Newton-Raphson Method\n');

    for iteration = 1:7
        augmentedMatrix = [laplaceMatrix - currentEigenvalue *
            eye(dimension), -initialVector;
                           -initialVector', 0];
        b1 = laplaceMatrix * initialVector - currentEigenvalue *
            initialVector;
        conditionNumber = cond(augmentedMatrix);
        norm_residual = norm(b1, 2);
        b = [b1; (1 - initialVector' * initialVector) / 2];
    endfor
endfunction

```

```

    solution = augmentedMatrix \ b;
    initialVector = initialVector - solution(1:dimension);
    currentEigenvalue = currentEigenvalue - solution(dimension + 1);
endfor

initialVector = originalVector;
printf('Simple Iteration with Shift\n');

for iteration = 1:5
    currentEigenvalue = initialVector' * laplaceMatrix *
    initialVector;
    modifiedMatrix = laplaceMatrix - currentEigenvalue *
    eye(dimension, dimension);
    conditionNumber = cond(modifiedMatrix);
    norm_residual = norm(modifiedMatrix * initialVector, 2);
    updatedVector = modifiedMatrix \ initialVector;
    initialVector = updatedVector / sqrt(updatedVector' *
    updatedVector);
endfor

endfunction

function laplaceMatrix = initializeLaplaceMatrix(gridSize)
    % Discrete Laplace operator in a square
    stepSize = 1 / gridSize;
    dimension = gridSize - 1;
    diagonalValues(1:dimension * dimension) = 4 / stepSize / stepSize;
    offDiagonalValues(1:dimension * dimension - 1) = -1 / stepSize /
    stepSize;

    for i = 1:dimension - 1
        offDiagonalValues(i * dimension) = 0;
    end
end

```

```
laplaceMatrix = diag(diagonalValues) +  
                  diag(offDiagonalValues, 1) +  
                  diag(offDiagonalValues, -1) +  
                  diag(offDiagonalValues, dimension) +  
                  diag(offDiagonalValues, -dimension);  
endfunction
```