

# Numerical Analysis

established about 4'000 years ago

- 1. Аналіз похибок заокруглення
  - 1.1. Види похибок
  - 1.2. Підрахунок похибок в ЕОМ
  - 1.3. Підрахунок похибок обчислення значення функції

## 1. Аналіз похибок заокруглення

Література:

- Самарский, Гулин, стор. 16–25: [djvu](#), [pdf](#);
- Березин, Жидков, том I, стор. 16–35: [djvu](#), [pdf](#).

### 1.1. Види похибок

Нехай необхідно розв'язати рівняння

$$Au = f. \quad (1)$$

За рахунок неточно заданих вхідних даних насправді ми маємо рівняння

$$\tilde{A}\tilde{u} = \tilde{f}. \quad (2)$$

**Означення:** Назвемо  $\delta_1 = u - \tilde{u}$  неусувною похибкою.

Застосування методу розв'язання (2) приводить до рівняння

$$\tilde{A}_h \tilde{u}_h = \tilde{f}_h, \quad (3)$$

де  $h > 0$  — малий параметр.

**Означення:** Назвемо  $\delta_2 = \tilde{u} - \tilde{u}_h$  похибкою методу.

Реалізація методу на ЕОМ приводить до рівняння

$$\tilde{A}_h^* \tilde{u}_h^* = \tilde{f}_h^*. \quad (4)$$

**Означення:** Назвемо  $\delta_3 = \tilde{u}_h - \tilde{u}_h^*$  похибкою заокруглення.

**Означення:** Тоді повна похибка

$$\delta = u - \tilde{u}_h^* = \delta_1 + \delta_2 + \delta_3.$$

**Означення:** кажуть, що задача (1) коректна, якщо

- $\forall f \in F: \exists! u \in U;$

- Задача (1) стійка, тобто  $\forall \varepsilon > 0: \exists \delta > 0:$

$$|A - \tilde{A}| < \delta, |f - \tilde{f}| < \delta \implies |u - \tilde{u}| < \varepsilon. \quad (5)$$

Якщо задача (1) некоректна, то або розв'язок її не існує, або він неєдиний, або він нестійкий, тобто  $\exists \varepsilon > 0: \forall \delta > 0:$

$$|A - \tilde{A}| < \delta, |f - \tilde{f}| < \delta \implies |u - \tilde{u}| > \varepsilon. \quad (6)$$

**Означення:** Абсолютна похибка  $\Delta x \leq |x - x^*|$ .

**Означення:** Відносна похибка  $\delta x \leq \Delta x / |x|$ , або  $\Delta x / |x^*|$ .

**Означення:** Значущими цифрами називаються всі цифри, починаючи з першої ненульової зліва.

**Означення:** Вірна цифра — це значуща, якщо абсолютна похибка за рахунок відкидання всіх молодших розрядів не перевищує одиниці розряду цієї цифри.

Тобто, якщо  $x^* = \alpha_n \dots \alpha_0. \alpha_{-1} \dots \alpha_{-p} \dots$ , то  $\alpha_{-p}$  вірна, якщо  $\Delta x \leq 10^{-p}$  (інколи  $\Delta x \leq w \cdot 10^{-p}$ , де  $1/2 \leq w < 1$  наприклад,  $w = 0.55$ ).

## 1.2. Підрахунок похибок в ЕОМ

Підрахуємо відносну похибку заокруглення числа  $x$  на ЕОМ з плаваючою комою. В  $\beta$ -ічній системі числення число представляється у вигляді

$$x = \pm(\alpha_1 \beta^{-1} + \alpha_2 \beta^{-2} + \dots + \alpha_t \beta^{-t} + \dots) \cdot \beta^p, \quad (7)$$

де  $0 \leq \alpha_k < \beta$ ,  $\alpha_1 \neq 0$ ,  $k = 1, 2, \dots$

Якщо в ЕОМ  $t$  розрядів, то при відкиданні молодших розрядів ми оперуємо з наближенням значенням

$$x^* = \pm(\alpha_1\beta^{-1} + \alpha_2\beta^{-2} + \dots + \alpha_t\beta^{-t}) \cdot \beta^p \quad (8)$$

і відповідно похибка заокруглення

$$x - x^* = \pm\beta^p \cdot (\alpha_{t+1}\beta^{-t-1} + \dots). \quad (9)$$

Тоді її можна оцінити так

$$\begin{aligned} |x - x^*| &\leq \beta^{p-t-1} \cdot (\beta - 1) \cdot (1 + \beta^{-1} + \dots) \leq \\ &\leq \beta^{p-t-1} \cdot (\beta - 1) \cdot \frac{1}{1 - \beta^{-1}} = \beta^{p-t}. \end{aligned} \quad (10)$$

Якщо в представлені (7) взяти  $\alpha_1 = 1$ , то  $|x| \geq \beta^p \cdot \beta^{-1}$ .

Звідси остаточно

$$\delta x \leq \frac{\beta^{p-t}}{\beta^{p-1}} = \beta^{-t+1}. \quad (11)$$

При більш точних способах заокруглення можна отримати оцінку  $\delta x \leq \frac{1}{2} \cdot \beta^{-t+1} = \varepsilon$ . Число  $\varepsilon$  називається «машинним іпсилоном». Наприклад, для  $\beta = 2$ ,  $t = 24$ ,  $\varepsilon = 2^{-24} \approx 10^{-7}$ .

### 1.3. Підрахунок похибок обчислення значення функції

Нехай задана функція  $y = f(x_1, \dots, x_n) \in C^1(\Omega)$ . Необхідно обчислити її значення при наближенному значенні аргументів

$\vec{x}^* = (x_1^*, \dots, x_n^*)$ , де  $|x_i - x_i^*| \leq \Delta x_i$  та оцінити похибку обчислення значення функції  $y^* = f(x_1^*, \dots, x_n^*)$ . Маємо

$$\begin{aligned} |y - y^*| &= |f(\vec{x}) - f(\vec{x}^*)| = \\ &= \left| \sum_{i=1}^n \frac{\partial f}{\partial x_i}(\vec{\xi}) \cdot (x_i - x_i^*) \right| \leq \\ &\leq \sum_{i=1}^n B_i \cdot \Delta x_i, \end{aligned} \quad (12)$$

$$\text{де } B_u = \max_{\vec{x} \in U} \left| \frac{\partial f}{\partial x_i}(\vec{x}) \right|.$$

Тут

$$U = \{ \vec{x} = (x_1, \dots, x_n) : |x_i - x_i^*| \leq \Delta x_i \} \subset \Omega, \quad (13)$$

для  $i = \overline{1, n}$ . Отже з точністю до величин першого порядку малості по

$$\Delta x = \max_i \Delta x_i, \quad (14)$$

$$\Delta y = |y - y^*| \prec \sum_{i=1}^n b_i \cdot \Delta x_i, \quad (15)$$

де  $b_i = \left| \frac{\partial f}{\partial x_i}(\vec{x}^*) \right|$  та « $\prec$ » означає приблизно менше.

Розглянемо похибки арифметичних операцій.

- Сума:  $y = x_1 + x_2$ ,  $x_1, x_2 > 0$ :

$$\Delta y \leq \Delta x_1 + \Delta x_2, \quad (16)$$

$$\delta y \leq \frac{\Delta x_1 + \Delta x_2}{x_1 + x_2} \leq \max(\delta x_1, \delta x_2). \quad (17)$$

- Різниця:  $y = x_1 - x_2$ ,  $x_1 > x_2 > 0$ :

$$\Delta y \leq \Delta x_1 + \Delta x_2, \quad (18)$$

$$\delta y \leq \frac{x_2 \delta x_1 + x_1 \delta x_2}{x_1 - x_2}. \quad (19)$$

При близьких  $x_1, x_2$  зростає відносна похибка (за рахунок втрати вірних цифр).

- Добуток:  $y = x_1 \cdot x_2$ ,  $x_1, x_2 > 0$ :

$$\Delta y \prec x_2 \Delta x_1 + x_1 \Delta x_2, \quad (20)$$

$$\delta y \leq \delta x_1 + \delta x_2. \quad (21)$$

- Частка  $y = x_1/x_2$ ,  $x_1, x_2 > 0$ :

$$\Delta y \prec \frac{x_2 \Delta x_1 + x_1 \Delta x_2}{x_2^2}, \quad (22)$$

$$\delta y \leq \delta x_1 + \delta x_2. \quad (23)$$

При малих  $x_2$  зростає абсолютна похибка (за рахунок зростання результату ділення).

**Означення:** Пряма задача аналізу похибок: обчислення  $\Delta y, \delta y$  по заданих  $\Delta x_i, i = \overline{1, n}$ .

**Означення:** Обернена задача: знаходження  $\Delta x_i$ ,  
 $i = \overline{1, n}$  по заданих  $\Delta y$ ,  $\delta y$ . Якщо  $n > 1$ , маємо одну умову

$$\sum_{i=1}^n b_i \cdot \Delta x_i < \varepsilon \quad (24)$$

для багатьох невідомих  $\Delta x_i$ .

Вибирають їх із однієї з умов:

$$\forall i : b_i \cdot \Delta x_i < \frac{\varepsilon}{n} \quad (25)$$

або

$$\Delta x_i < \frac{\varepsilon}{\sum_{i=1}^n b_i}. \quad (26)$$

[Назад до лекцій](#)

[Назад на головну](#)

---

**numerical-analysis is maintained by csc-knu.**

© 2019 Київський національний університет імені Тараса Шевченка, Андрій Риженко, Скибицький Нікіта

# Numerical Analysis

established about 4'000 years ago

- 2. Методи розв'язання нелінійних рівнянь
  - 2.1. Метод ділення навпіл
  - 2.2. Метод простої ітерації
  - 2.3. Метод релаксації
  - 2.4. Метод Ньютона (метод дотичних)
  - 2.5. Збіжність методу Ньютона

## 2. Методи розв'язання нелінійних рівнянь

Постановка задачі. Нехай маємо рівняння  $f(x) = 0$ ,  $\bar{x}$  — його розв'язок, тобто  $f(\bar{x}) = 0$ .

Задача розв'язання цього рівняння розпадається на етапи:

- Існування та кількість коренів.
- Віddілення коренів, тобто розбиття числової вісі на інтервали, де знаходиться один корінь.
- Обчислення кореня із заданою точністю  $\varepsilon$ .

Для розв'язання перших двох задач використовуються методи математичного аналізу та алгебри, а також графічний метод. Далі розглядаються методи розв'язання третього етапу.

### 2.1. Метод ділення навпіл

Література:

- Самарський, Гулин, стор. 191: [djvu](#), [pdf](#);
- Волков, стор. 189–190: [djvu](#), [pdf](#).

Припустимо на  $[a, b]$  знаходиться лише один корінь рівняння

$$f(x) = 0 \tag{1}$$

для  $f(x) \in C[a, b]$ , який необхідно визначити. Нехай  $f(a) \cdot f(b) < 0$ .

Припустимо, що  $f(a) > 0$ ,  $f(b) < 0$ . Покладемо  $x_1 = \frac{a+b}{2}$  і підрахуємо  $f(x_1)$ . Якщо  $f(x_1) < 0$ , тоді шуканий корінь  $\bar{x}$  знаходиться на інтервалі  $(a, x_1)$ . Якщо ж  $f(x_1) > 0$ , то  $\bar{x} \in (x_1, b)$ . Далі з двох інтервалів  $(a, x_1)$  і  $(x_1, b)$  вибираємо той, на границях якого функція  $f(x)$  має різні знаки, знаходимо точку  $x_2$  — середину вибраного інтервалу, підраховуємо  $f(x_2)$  і повторюємо вказаний процес.

В результаті отримаємо послідовність інтервалів, що містять шуканий корінь  $\bar{x}$ , причому довжина кожного послідувального інтервалу вдвічі менше попереднього.

Цей процес продовжується до тих пір, поки довжина отриманого інтервалу  $(a_n, b_n)$  не стане меншою за  $b_n - a_n < 2\varepsilon$ . Тоді  $x_{n+1}$ , як середина інтервалу  $(a_n, b_n)$ , пов'язане з  $\bar{x}$  нерівністю

$$|x_{n+1} - \bar{x}| < \varepsilon. \quad (2)$$

Ця умова для деякого  $n$  буде виконуватись за теоремою Больцано-Коши. Оскільки

$$|b_{k+1} - a_{k+1}| = \frac{|b_k - a_k|}{2}, \quad (3)$$

то

$$|x_{n+1} - \bar{x}| \leq \frac{b - a}{2^{n+1}} < \varepsilon. \quad (4)$$

Звідси отримаємо нерівність для обчислення кількості ітерацій  $n$  для виконання умови (2):

$$n = n(\varepsilon) \geq \left\lceil \log\left(\frac{b - a}{\varepsilon}\right) \right\rceil + 1. \quad (5)$$

Степінь збіжності — лінійна, тобто геометричної прогресії з знаменником  $q = 1/2$ .

- **Переваги методу:** простота, надійність.
- **Недоліки методу:** низька швидкість збіжності; метод не узагальнюється на системи.

## 2.2. Метод простої ітерації

Література:

- Самарський, Гулин, стор. 191–193: [djvu](#), [pdf](#);
- Волков, стор. 172–184: [djvu](#), [pdf](#).

Спочатку рівняння

$$f(x) = 0 \quad (6)$$

замінюється еквівалентним

$$x = \varphi(x). \quad (7)$$

Ітераційний процес має вигляд

$$x_{n+1} = \varphi(x_n), \quad n = 0, 1, \dots \quad (8)$$

Початкове наближення  $x_0$  задається.

Для збіжності велике значення має вибір функції  $\varphi(x)$ . Перший спосіб заміни рівняння полягає в віддаленні змінної з якогось члена рівняння. Більш продуктивним є перехід від рівняння (6) до (7) з функцією  $\varphi(x) = x + \tau(x) \cdot f(x)$ , де  $\tau(x)$  — знакостала функція на тому відрізку, де шукаємо корінь.

**Означення:** Кажуть, що ітераційний метод збігається, якщо  $\lim_{k \rightarrow \infty} x_k = \bar{x}$ .

Далі  $U_r = \{x : |x - a| \leq r\}$  відрізок довжини  $2r$  з серединою в точці  $a$ .

З'ясуємо умови, при яких збігається метод простої ітерації.

**Теорема 1:** Якщо

$$\max_{x \in [a, b] \cap U_r} |\varphi'(x)| \leq q < 1 \quad (9)$$

то метод простої ітерації збігається і має місце оцінка

$$|x_n - \bar{x}| \leq \frac{q^n}{1-q} \cdot |x_0 - x_1| \leq \frac{q^n}{1-q} \cdot (b - a). \quad (10)$$

*Доведення:* Нехай  $x_{k+1}, x_k \in U_r$ . Тоді

$$\begin{aligned} |x_k - x_{k-1}| &= |\varphi(x_k) - \varphi(x_{k-1})| = \\ &= |\varphi'(\xi_k) \cdot (x_k - x_{k-1})| \leq \\ &\leq |\varphi'(\xi_k)| \cdot |x_k - x_{k-1}| \leq \\ &\leq q \cdot |x_k - x_{k-1}| = \dots \\ &= q^k \cdot |x_1 - x_0|, \end{aligned} \quad (11)$$

де  $\xi_k = x_k + \theta_k \cdot (x_{k+1} - x_k)$ , а у свою чергу  $0 < \theta_k < 1$ . Далі

$$\begin{aligned} |x_{k+p} - x_k| &= |x_{k+p} - x_{k+p-1} + \dots + x_{k+1} - x_k| = \\ &= |x_{k+p} - x_{k+p-1}| + \dots + |x_{k+1} - x_k| \leq \\ &\leq (q^{k+p-1} + q^{k+p-2} + \dots + q^k) \cdot |x_1 - x_0| = \\ &= \frac{q^k - q^{k+p-1}}{1-q} \cdot |x_1 - x_0| \xrightarrow{k \rightarrow \infty} 0. \end{aligned} \quad (12)$$

Бачимо що  $\{x_k\}$  — фундаментальна послідовність. Значить вона збіжна. При  $p \rightarrow \infty$  в (12) отримуємо (10).  $\square$

Визначимо кількість ітерацій для досягнення точності  $\varepsilon$ . З оцінки в теоремі 1 отримаємо

$$|x_n - \bar{x}| \leq \frac{q^n}{1-q} \cdot (b - a) < \varepsilon, \quad (13)$$

звідки безпосередньо маємо

$$n(\varepsilon) = n \geq \left\lceil \frac{\ln\left(\frac{\varepsilon(1-q)}{b-a}\right)}{\ln q} \right\rceil + 1. \quad (14)$$

Практично ітераційний процес зупиняємо при:  $|x_n - x_{n-1}| < \varepsilon$ . Але ця умова не завжди гарантує, що  $|x_n - \bar{x}| < \varepsilon$ .

**Зауваження:** Умова збіжності методу може бути замінена на умову Ліпшица

$$|\varphi(x) - \varphi(y)| \leq q \cdot |x - y|, \quad 0 < q < 1. \quad (15)$$

- **Переваги методу:** простота; при  $q < 1/2$  — швидше збігається ніж метод ділення навпіл; метод узагальнюється на системи.
- **Недоліки методу:** при  $q > 1/2$  збігається повільніше ніж метод ділення навпіл; виникають труднощі при зведенні  $f(x) = 0$  до  $x = \varphi(x)$ .

## 2.3. Метод релаксації

Література:

- Самарський, Гулин, стор. 192–193: [djvu](#), [pdf](#).

Якщо в методі простої ітерації для рівняння  $x = x + \tau \cdot f(x) \equiv \varphi(x)$  вибрati  $\tau(x) = \tau = \text{const}$ , то ітераційний процес приймає вигляд

$$x_{n+1} = x_n + \tau \cdot f(x_n), \quad (16)$$

де  $k = 0, 1, 2, 3 \dots$ , а  $x_0$  — задано. Метод можна записати у вигляді

$$\frac{x_{k+1} - x_k}{\tau} = f(x_k), \quad k = 0, 1, \dots \quad (17)$$

Оскільки  $\varphi'(x) = 1 + \tau \cdot f'(x)$ , то метод збігається при умові

$$|\varphi'(x)| = |1 + \tau \cdot f'(x)| \leq q < 1. \quad (18)$$

Нехай  $f'(x) < 0$ , тоді (8) запишеться у вигляді:  $-q \leq 1 + \tau \cdot f'(x) \leq q < 1$ . Звідси

$$f'(x) \leq 1 + q < 2k\tau, \quad (19)$$

i

$$0 < \tau < \frac{2}{|f'(x)|}. \quad (20)$$

Поставимо задачу знаходження  $\tau$ , для якого  $q = q(\tau) \rightarrow \min$ . Для того, щоб вибирати оптимальний параметр  $\tau$ , розглянемо рівняння для похиби  $z_k = x_k - \bar{x}$ .

Підставивши  $x_k = x + z_k$  в (16), отримаємо

$$z_{k+1} = z_k + \tau \cdot f(x + z_k). \quad (21)$$

В припущені  $f(x) \in C^1([a, b])$  з теореми про середнє маємо

$$\begin{aligned} f(\bar{x} + z_k) &= f(\bar{x}) + z_k \cdot f'(\bar{x} + \theta \cdot z_k) = \\ &= z_k \cdot f'(\bar{x} + \theta \cdot z_k) = z_k \cdot f'(\xi_k), \end{aligned} \quad (22)$$

тобто

$$z_{k+1} = z_k + \tau \cdot f'(\xi_k) \cdot z_k. \quad (23)$$

Звідси

$$|z_{k+1}| \leq |1 + \tau \cdot f'(\xi_k)| \cdot |z_k| \leq \max_U |1 + \tau \cdot f'(\xi_k)| \cdot |z_k|. \quad (24)$$

А тому

$$|z_{k+1}| \leq \max \{|1 - \tau M_1|, |1 - \tau m_1|\} \cdot |z_k|, \quad (25)$$

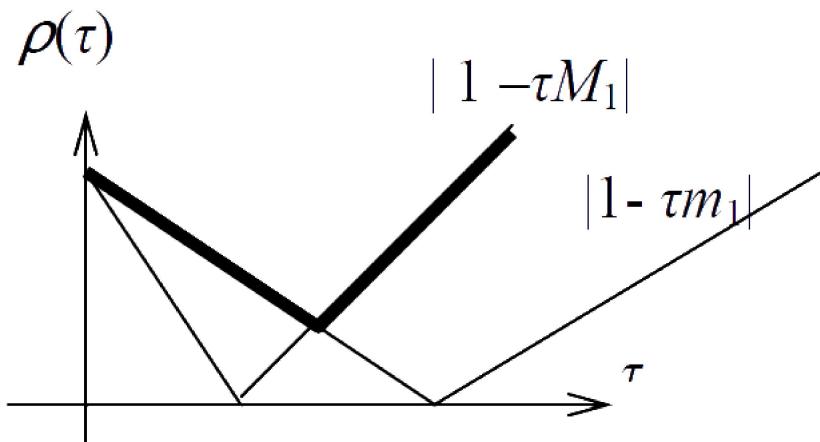
де

$$m_1 = \min_{[a,b]} |f'(x)|, \quad M_1 = \max_{[a,b]} |f'(x)| \quad (26)$$

Таким чином, задача вибору оптимального параметра зводиться до знаходження  $\tau$ , для якого функція

$$q(\tau) = \max \{|1 - \tau M_1|, |1 - \tau m_1|\} \quad (27)$$

приймає мінімальне значення:  $q(\tau) \rightarrow \min$ .



З графіка видно, що точка мінімуму визначається умовою  $|1 - \tau M_1| = |1 - \tau m_1|$ . Тому

$$1 - \tau_0 m_1 = \tau_0 M_1 - 1 \implies \tau_0 = \frac{2}{M_1 - m_1} < \frac{2}{|f'(x)|}. \quad (28)$$

При цьому значенні  $\tau$  маємо

$$q(\tau_0) = \rho_0 = \frac{M_1 - m_1}{M_1 + m_1}. \quad (29)$$

Тоді для похибки вірна оцінка

$$|x_n - \bar{x}| \leq \frac{\rho_0^n}{1 - \rho_0} \cdot (b - a) < \varepsilon. \quad (30)$$

Кількість ітерацій

$$n = n(\varepsilon) \geq \left\lceil \frac{\frac{\ln(\varepsilon(1-\rho_0))}{b-a}}{\ln \rho_0} \right\rceil + 1. \quad (31)$$

**Задача 1:** Дати геометричну інтерпретацію методу простої ітерації для випадків:

$$0 < \varphi'(x) < 1; \quad -1 < \varphi'(x) < 0; \quad \varphi'(x) < -1; \quad \varphi'(x) > 1. \quad (32)$$

**Задача 2:** Знайти оптимальне  $\tau = \tau_0$  для методу релаксації при  $f'(x) > 0$ .

## 2.4. Метод Ньютона (метод дотичних)

Література:

- Самарський, Гулин, стор. 193–194: [djvu](#), [pdf](#).
- Березин, Жидков, том. II, стор. 135–140: [djvu](#), [pdf](#).

Припустимо, що рівняння  $f(x) = 0$  має простий дійсний корінь  $\bar{x}$ , тобто  $f(\bar{x}) = 0$ ,  $f'(x) \neq 0$ . Нехай виконуються умови:  $f(x) \in C^1([a, b])$ ,  $f(a) \cdot f(b) < 0$ . Тоді

$$0 = f(\bar{x}) = f(x_k + \bar{x} - x_k) = f(x_k) + f'(\xi_k) \cdot (x - x_k), \quad (33)$$

де  $\xi_k = x_k + \theta_k \cdot (\bar{x} - x_k)$ ,  $0 < \theta_k < 1$ ,  $\xi_k \approx x_k$ . Тому наступне наближення виберемо з рівняння

$$f(x_k) + f'(x_k) \cdot (x_{k+1} - x_k) = 0. \quad (34)$$

Звідси маємо ітераційний процес

$$x_{k+1} = x_k - \frac{f(x_k)}{f'(x_k)}, \quad (35)$$

де  $k = 0, 1, 2, \dots$ ;  $x_0$  — задане.

Метод Ньютона ще називають методом лінеаризації або методом дотичних.

**Задача 3:** Дати геометричну інтерпретацію методу Ньютона.

Метод Ньютона можна інтерпретувати як метод простої ітерації з

$$\varphi(x) = x - \frac{f(x)}{f'(x)}, \quad (36)$$

тобто

$$\tau(x) = -\frac{1}{f'(x)}. \quad (37)$$

Тому

$$\varphi'(x) = 1 - \frac{f'(x) \cdot f'(x) - f(x) \cdot f''(x)}{(f'(x))^2} = \frac{f(x) \cdot f''(x)}{(f'(x))^2}. \quad (38)$$

Якщо  $\bar{x}$  — корінь  $f(x)$ , то  $\varphi'(x) = 1$ . знайдеться окіл кореня,

$$|\varphi'(x)| = \left| \frac{f(x) \cdot f''(x)}{(f'(x))^2} \right| < 1. \quad (39)$$

Це означає, що збіжність методу Ньютона залежить від вибору  $x_0$ .

**Недолік** методу Ньютона: необхідність обчислювати на кожній ітерації не тільки значення функції, а й похідної.

Модифікований метод Ньютона позбавлений цього недоліку і має вигляд:

$$x_{k+1} = x_k - \frac{f(x_k)}{f'(x_0)}, \quad k = 0, 1, 2, \dots \quad (40)$$

Цей метод має лише лінійну збіжність:  $|x_{k+1} - x| = O(|x_k - \bar{x}|)$ .

**Задача 4:** Дати геометричну інтерпретацію модифікованого методу Ньютона.

В методі Ньютона, для якого  $f'(x_k)$  замінюється на

$$\frac{f(x_k) - f(x_{k-1})}{x_k - x_{k-1}} \quad (41)$$

дає метод січних:

$$x_{k+1} = x_k - \frac{x_k - x_{k-1}}{f(x_k) - f(x_{k-1})} \cdot f(x_k), \quad (42)$$

де  $k = 1, 2, \dots, x_0, x_1$  — задані.

**Задача 5:** Дати геометричну інтерпретацію методу січних.

## 2.5. Збіжність методу Ньютона

Література:

- Самарський, Гулин, стор. 199–202: [djvu](#), [pdf](#).

**Теорема 1:** Нехай  $f(x) \in C^2([a, b])$ ;  $\bar{x}$  простий дійсний корінь рівняння

$$f(x) = 0. \quad (43)$$

і  $f'(x) \neq 0$  при  $x \in U_r = \{x : |x - \bar{x}| < r\}$ . Якщо

$$q = \frac{M_2 \cdot |\bar{x} - x_0|}{2m_1} < 1, \quad (44)$$

де

$$m_1 = \min_{U_r} |f'(x)|, \quad M_2 = \max_{U_r} |f''(x)|, \quad (45)$$

то для  $x_0 \in U_r$  метод Ньютона

$$x_{k+1} = x_k - \frac{f(x_k)}{f'(x_k)} \quad (46)$$

збігається і має місце оцінка

$$|x_n - \bar{x}| \leq q^{2^n - 1} \cdot |x_0 - \bar{x}|. \quad (47)$$

З (46) маємо

$$\begin{aligned} x_{k+1} - \bar{x} &= x_k - \frac{f(x_k)}{f'(x_k)} - \bar{x} = \\ &= \frac{(x_k - \bar{x}) \cdot f'(x_k) - f(x_k)}{f'(x_k)} = \frac{F(x_k)}{f'(x_k)}, \end{aligned} \quad (48)$$

де  $F(x) = (x - \bar{x})f'(x) - f(x)$ , така, що

- $F(\bar{x}) = 0$ ;
- $F'(x) = (x - \bar{x}) \cdot f''(x)$ .

Тоді

$$F(x_k) = F(\bar{x}) + \int_x^{x_k} F'(t) dt = \int_x^{x_k} (t - \bar{x}) \cdot f''(t) dt. \quad (49)$$

Так як  $(t - \bar{x})$  не міняє знак на відрізку інтегрування, то скористаємося теоремою про середнє значення:

$$F(x_k) = f''(\xi_k) \int_x^{x_k} (t - \bar{x}) dt = \frac{(x_k - x)^2}{2} \cdot f''(\xi_k), \quad (50)$$

де  $\xi_k = \bar{x} + \theta_k \cdot (x_k - \bar{x})$ , де  $0 < \theta_k < 1$ . З (48), (50) маємо

$$x_{k+1} - \bar{x} = \frac{(x_k - \bar{x})^2}{2f'(x_k)} \cdot f''(\xi_k). \quad (51)$$

Доведемо оцінку (46) за індукцією. Так як  $x_0 \in U_r$ , то

$$|\xi_0 - \bar{x}| = |\theta_0 \cdot (x_0 - \bar{x})| < |\theta_0| \cdot |x_0 - \bar{x}| < r \quad (52)$$

звідси випливає  $\xi_0 \in U_r$ .

Тоді  $f''(\xi_0) \leq M_2$ , тому

$$|x_1 - \bar{x}| \leq \frac{(x_0 - \bar{x})^2 \cdot M_2}{2m_1} = \frac{M_2 \cdot |x_0 - \bar{x}|}{2m_1} \cdot |x_0 - \bar{x}| = q \cdot |x_0 - \bar{x}| < r,$$

тобто  $x_1 \in U_r$ .

Ми довели твердження (47) при  $n = 1$ . Нехай воно справджується при  $n = k$

$$\begin{aligned} |x_k - \bar{x}| &\leq q^{2^{k-1}} \cdot |x_0 - \bar{x}| < r, \\ |\xi_k - \bar{x}| &= |\theta_k \cdot (x_k - \bar{x})| < r. \end{aligned} \quad (54)$$

Тоді  $x_k, \xi_k \in U_r$ .

Доведемо (47) для  $n = k + 1$ . З (51) маємо

$$\begin{aligned} |x_{k+1} - \bar{x}| &\leq \frac{|x_k - \bar{x}|^2 \cdot M_2}{2m_1} \leq \\ &\leq \left(q^{2^{k-1}}\right)^2 \cdot \frac{|x_0 - \bar{x}|^2 \cdot M_2}{2m_1} = \\ &= q^{2^{k+1}-2} \cdot \frac{|x_0 - \bar{x}| \cdot M_2}{2m_1} \cdot |x_0 - \bar{x}| = \\ &= q^{2^{k+1}-1} \cdot |x_0 - \bar{x}|. \end{aligned}$$

Таким чином (47) справджується для  $n = k + 1$ . Значить (47) виконується і для довільного  $n$ . Таким чином  $x_n \xrightarrow[n \rightarrow \infty]{} x$ .  $\square$

З (47) маємо оцінку кількості ітерацій для досягнення точності  $\varepsilon$

$$n \geq \left\lceil \log_2 \left( 1 + \frac{\ln \left( \frac{\varepsilon}{b-a} \right)}{\ln q} \right) \right\rceil + 1. \quad (55)$$

Кажуть, що ітераційний метод має *степінь збіжності*  $m$ , якщо

$$|x_{k+1} - \bar{x}| = O(|x_k - \bar{x}|^m). \quad (56)$$

Для методу Ньютона

$$|x_{k+1} - \bar{x}| = \frac{|x_k - \bar{x}|^2 \cdot |f''(\xi_k)|}{2|f'(x_k)|}. \quad (57)$$

Звідси випливає, що

$$|x_{k+1} - \bar{x}| = O(|x_k - \bar{x}|^2). \quad (58)$$

Значить степінь збіжності методу Ньютона  $m = 2$ . Для методу простої ітерації і ділення навпіл  $m = 1$ .

**Теорема 2:** Нехай  $f(x) \in C^2([a, b])$  та  $x$  простий корінь рівняння  $f(x) = 0$  ( $f'(x) \neq 0$ ). Якщо  $f'(x) \cdot f''(x) > 0$  ( $f'(x) \cdot f''(x) < 0$ ) то для методу Ньютона при  $x_0 = b$  послідовність наближень  $\{x_k\}$  монотонно спадає (монотонно зростає при  $x_0 = a$ ).

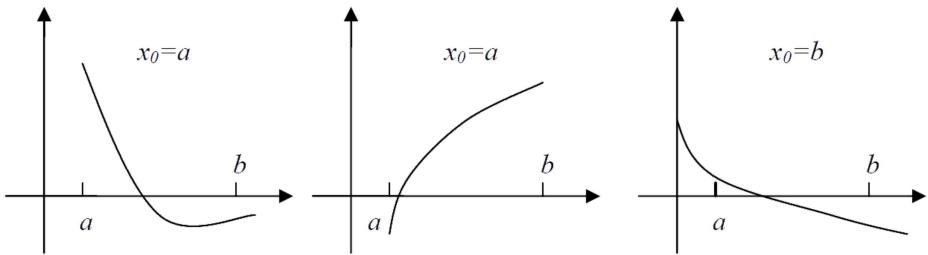
**Задача 6:** Довести теорему 2 при

- $f'(x) \cdot f''(x) > 0$ ;
- $f'(x) \cdot f''(x) < 0$ .

**Задача 7:** Знайти степінь збіжності методу січних [Каліткін Н.Н., Численные методы, с. 145–146]

Якщо  $f(a) \cdot f''(a) > 0$  та  $f''(x)$  не міняє знак, то потрібно вибирати  $x_0 = a$ ; при цьому  $\{x_k\} \uparrow \bar{x}$ .

Якщо  $f(b) \cdot f''(b) > 0$ , то  $x_0 = b$ ; маємо  $\{x_k\} \downarrow \bar{x}$ . Пояснення на рисунку 2:



**Зауваження 1:** Якщо  $\bar{x}$  —  $p$ -кратний корінь тобто

$$f^{(m)}(\bar{x}) = 0, \quad m = 0, 1, \dots, p-1; \quad f^{(p)}(\bar{x}) \neq 0, \quad (59)$$

то в методі Ньютона необхідна наступна модифікація

$$x_{k+1} - x_k - p \cdot \frac{f(x_k)}{f'(x_k)} \quad (60)$$

i

$$q = \frac{M_{p+1} \cdot |x_0 - \bar{x}|}{m_p \cdot p \cdot (p+1)} < 1. \quad (61)$$

**Зауваження 2:** Метод Ньютона можна застосовувати і для обчислення комплексного кореня

$$z_{k+1} = z_k - \frac{f(z_k)}{f'(z_k)} \quad (62)$$

В теоремі про збіжність

$$q = \frac{|x_0 - \bar{x}| M_2}{2m_1}, \quad (63)$$

де

$$m_1 = \min_{U_r} |f'(z)|, \quad M_2 = \max_{U_r} |f''(z)|. \quad (64)$$

Тут  $|z|$  — модуль комплексного числа.

**Переваги** методу Ньютона:

- висока швидкість збіжності;
- узагальнюється на системи рівнянь;

- узагальнюється на комплексні корені.

**Недоліки** методу Ньютона:

- на кожній ітерації обчислюється не тільки  $f(x_k)$ , а і похідна  $f'(x_k)$ ;
- збіжність залежить від початкового наближення  $x_0$ , оскільки від нього залежить умова збіжності

$$q = \frac{M_2|x_0 - \bar{x}|}{2m_1} < 1; \quad (65)$$

- потрібно, щоб  $f(x) \in C^2([a, b])$ .

[Назад до лекцій](#)

[Назад на головну](#)

---

**numerical-analysis** is maintained by [csc-knu](#).

© 2019 Київський національний університет імені Тараса Шевченка, Андрій Риженко,  
Скибицький Нікіта

# Numerical Analysis

established about 4'000 years ago

- 3. Методи розв'язання систем лінійних алгебраїчних рівнянь (СЛАР)
  - 3.1. Метод Гаусса
  - 3.2. Метод квадратних коренів
  - 3.3. Обчислення визначника та оберненої матриці
  - 3.4. Метод прогонки
  - 3.5. Обумовленість систем лінійних алгебраїчних рівнянь

## 3. Методи розв'язання систем лінійних алгебраїчних рівнянь (СЛАР)

Методи розв'язування СЛАР поділяються на *прямі* та *ітераційні*. При умові точного виконання обчислень прямі методи за скінчену кількістю операцій в результаті дають точний розв'язок.

Використовуються вони для невеликих та середніх СЛАР  $n = 10^2 - 10^4$ . Ітераційні методи використовуються для великих СЛАР  $n > 10^5$ , як правило розріджених. В результаті отримуємо послідовність наближень, яка збігається до розв'язку.

### 3.1. Метод Гаусса

Література:

- Самарский, Гулин, стор. 49–67: [djvu](#), [pdf](#);
- Березин, Жидков, том II, стор. 10–17: [djvu](#), [pdf](#).

Розглянемо задачу розв'язання СЛАР

$$A\vec{x} = \vec{b}, \quad (1)$$

причому  $A = (a_{ij})_{i,j=1}^n$ ,  $\det A \neq 0$ ,  $\vec{x} = (x_i)_{i=1}^n$ ,  $\vec{b} = (b_j)_{j=1}^n$ . Метод Крамера з обчисленням визначників для такої системи має складність  $Q = O(n! \cdot n)$ .

Запишемо СЛАР у вигляді

$$\begin{cases} a_{1,1}x_1 + a_{1,2}x_2 + \dots + a_{1,n}x_n = b_1 \equiv a_{1,n+1}, \\ a_{2,1}x_1 + a_{2,2}x_2 + \dots + a_{2,n}x_n = b_2 \equiv a_{2,n+1}, \\ \dots \\ a_{n,1}x_1 + a_{n,2}x_2 + \dots + a_{n,n}x_n = b_n \equiv a_{n,n+1}. \end{cases} \quad (2)$$

Якщо  $a_{1,1} \neq 0$ , то ділимо перше рівняння на нього і виключаємо  $x_1$  з інших рівнянь:

$$\begin{cases} x_1 + a_{1,2}^{(1)}x_2 + \dots + a_{1,n}^{(1)}x_n = a_{1,n+1}^{(1)}, \\ a_{2,2}^{(1)}x_2 + \dots + a_{2,n}^{(1)}x_n = a_{2,n+1}^{(1)}, \\ \dots \\ a_{n,2}^{(1)}x_2 + \dots + a_{n,n}^{(1)}x_n = a_{n,n+1}^{(1)}. \end{cases} \quad (3)$$

Процес повторюємо для  $x_2, \dots, x_n$ . В результаті отримуємо систему з трикутною матрицею

$$\begin{cases} x_1 + a_{1,2}^{(1)}x_2 + \dots + a_{1,n}^{(1)}x_n = a_{1,n+1}^{(1)}, \\ x_2 + \dots + a_{2,n}^{(2)}x_n = a_{2,n+1}^{(2)}, \\ \dots \\ x_n = a_{n,n+1}^{(n)}. \end{cases} \quad (4)$$

Тобто

$$A^{(n)} \vec{x} = \vec{a}^{(n)}. \quad (5)$$

Це прямий хід методу Гаусса. Формули прямого ходу

```
for k in range(1, n):
    for j in range(k + 1, n + 2):
        a[k, j][k] = a[k, j][k - 1] / a[k, k][k - 1]
    for i in range(k + 1, n + 1):
        a[i, j][k] = a[i, j][k - 1] - \
            a[i, j][k - 1] * a[k, j][k]
```

Звідси

$$x_n = a_{n,n+1}^{(n)}, \quad x_i = a_{i,n+1}^{(i)} - \sum_{j=i+1}^n a_{i,j}^{(n)} x_j, \quad (6)$$

для  $i = \overline{n-1, 1}$ . Це формули оберненого ходу.

Складність, тобто кількість операцій, яку необхідно виконати для реалізації методу:  $Q_f = 2/3n^2 + O(n^2)$  для прямого ходу,  $Q_b = n^2 + O(n)$  для оберненого ходу.

Умова

$$a_{k,k}^{(k-1)} \neq 0 \quad (7)$$

не суттєва, оскільки знайдеться  $m$ , для якого

$$\left| a_{m,k}^{(k-1)} \right| = \max_i \left| a_{i,k}^{(k-1)} \right| \neq 0 \quad (8)$$

(оскільки  $\det A \neq 0$ ). Тоді міняємо місцями рядки номерів  $k$  і  $m$ .

**Означення:** Елемент

$$a_{k,k}^{(k-1)} \neq 0 \quad (9)$$

називається *ведучим*.

Введемо матриці

$$M_k = \begin{pmatrix} 1 & \cdots & 0 & \cdots & 0 \\ \vdots & \ddots & \vdots & \ddots & \vdots \\ 0 & \cdots & m_{k,k} & \cdots & 0 \\ \vdots & \ddots & \vdots & \ddots & \vdots \\ 0 & \cdots & m_{n,k} & \cdots & 1 \end{pmatrix} \quad (10)$$

елементи якої обчислюються так:

$$m_{k,k} = \frac{1}{a_{k,k}^{(k-1)}}, \quad m_{k,k} = -\frac{a_{i,k}^{(k-1)}}{a_{k,k}^{(k-1)}}. \quad (11)$$

Нехай на  $k$ -му кроці  $A_{k-1}\vec{x} = \vec{b}_{k-1}$ . Множимо цю СЛАР зліва на  $M_k$ :  $M_k A_{k-1}\vec{x} = M_k \vec{b}_{k-1}$ . Позначимо  $A_k = M_k A_{k-1}$ ;  $A_0 = A$ . Тоді прямий хід методу Гаусса можна записати у вигляді

$$M_n M_{n-1} \dots M_1 A \vec{x} = M_n M_{n-1} \dots M_1 \vec{b}. \quad (12)$$

Позначимо останню систему, яка співпадає з (5), так

$$U \vec{x} = \vec{c}, \quad U = (u_{i,j})_{i,j=1}^n, \quad (13)$$

причому

$$\begin{cases} u_{i,i} = 1, \\ u_{i,j} = 0, \quad i > j. \end{cases} \quad (14)$$

Таким чином  $U = M_n M_{n-1} \dots M_1 A$ . Введемо матриці

$$L_k = M_k^{-1} = \begin{pmatrix} 1 & \cdots & 0 & \cdots & 0 \\ \vdots & \ddots & \vdots & \ddots & \vdots \\ 0 & \cdots & a_{k,k}^{(k-1)} & \cdots & 0 \\ \vdots & \ddots & \vdots & \ddots & \vdots \\ 0 & \cdots & a_{n,k}^{(k-1)} & \cdots & 1 \end{pmatrix} \quad (15)$$

Тоді

$$A = L_1 \dots L_n U = LU; \quad L = L_1 \dots L_n, \quad (16)$$

де  $L$  — нижня трикутна матриця,  $U$  — верхня трикутна матриця. Таким чином метод Гаусса можна трактувати, як розклад матриці  $A$  в добуток двох трикутних матриць —  $LU$ -розклад.

Введемо матрицю перестановок на  $k$ -му кроці (це матриця, отримана з одиничної матриці перестановкою  $k$ -того і  $m$ -того рядка). Тоді при множенні на неї матриці  $A_{k-1}$  робимо ведучим елементом максимальний за модулем.

$$P_k = \begin{pmatrix} 1 & \cdots & 0 & \cdots & 0 & \cdots & 0 \\ \vdots & \ddots & \vdots & \ddots & \vdots & \ddots & \vdots \\ 0 & \cdots & 0 & \cdots & 1 & \cdots & 0 \\ \vdots & \ddots & \vdots & \ddots & \vdots & \ddots & \vdots \\ 0 & \cdots & 1 & \cdots & 0 & \cdots & 0 \\ \vdots & \ddots & \vdots & \ddots & \vdots & \ddots & \vdots \\ 0 & \cdots & 0 & \cdots & 0 & \cdots & 1 \end{pmatrix} \quad (17)$$

За допомогою цих матриць перехід до трикутної системи (13) тепер має вигляд:

$$M_n M_{n-1} P_{n-1} \dots M_1 P_1 A \vec{x} = M_n M_{n-1} P_{n-1} \dots M_1 P_1 \vec{b}. \quad (18)$$

**Твердження:** Знайдеться така матриця  $P$  перестановок, що  $PA = LU$  — розклад матриці на нижню трикутну з ненульовими діагональними елементами і верхню трикутну матрицю з одиницями на діагоналі.

Висновки про **переваги** трикутного розкладу:

- Розділення прямого і оберненого ходів дає змогу економно розв'язувати декілька систем з одноковою матрицею та різними правими частинами.
- Зберігання  $M$ , або  $L$  та  $U$  на місці  $A$ .
- Обчислюючи  $\ell$  — кількість перестановок, можна встановити знак визначника.

### 3.2. Метод квадратних коренів

Література:

- Самарський, Гулин, стор. 69–73: [djvu](#), [pdf](#);
- Березин, Жидков, том II, стор. 23–25: [djvu](#), [pdf](#).

Цей метод призначений для розв'язання систем рівнянь із симетричною матрицею

$$A \vec{x} = \vec{b}, \quad A^\top = A. \quad (19)$$

Він оснований на розкладі матриці  $A$  в добуток:

$$A = S^\top D S, \quad (20)$$

де  $S$  — верхня трикутна матриця,  $S^\top$  — нижня трикутна матриця,  $D$  — діагональна матриця.

Виникає питання: як обчислити  $S, D$  по матриці  $A$ ? Маємо

$$DS_{i,j} = \begin{cases} d_{i,i}s_{i,j}, & i \leq j, \\ 0, & i > j. \end{cases} \quad (21)$$

Далі

$$\begin{aligned} S^\top DS_{i,j} &= \sum_{l=1}^n s_{i,l}^\top d_{l,l} s_{l,j} = \\ &= \sum_{l=1}^{i-1} s_{l,i}^\top s_{l,j} d_{l,l} + s_{i,i} s_{i,j} d_{i,i} + \\ &\quad + s_{l,i}^\top \underbrace{\sum_{l=i+1}^n s_{l,i}^\top s_{l,j} d_{l,l}}_{=0} = a_{i,j}, \end{aligned} \quad (22)$$

для  $i, j = \overline{1, n}$ .

Якщо  $i = j$ , то

$$|s_{i,i}^2|d_{i,i} = a_{i,i} - \sum_{l=1}^{i-1} |s_{l,i}^2|d_{l,l} \equiv p_i. \quad (23)$$

Тому

$$d_{i,i} = \text{sign}(p_i), \quad s_{i,i} = \sqrt{|p_i|}. \quad (24)$$

Якщо  $i < j$ , то

$$s_{i,j} = \left( a_{i,j} - \sum_{l=1}^{i-1} s_{l,i}^\top d_{l,l} s_{l,j} \right) / (s_{i,i} d_{i,i}), \quad (25)$$

де  $i = \overline{1, n}$ , а  $j = \overline{i+1, n}$ .

Якщо  $A > 0$  (тобто головні мінори матриці  $A$  додатні), то всі  $d_{i,i} = +1$ .

Знайдемо розв'язок рівняння (19). Враховуючи (20), маємо:

$$S^\top D \vec{y} = \vec{b} \quad (26)$$

|

$$S \vec{x} = \vec{y} \quad (27)$$

Оскільки  $S$  — верхня трикутна матриця, а  $S^\top D$  — нижня трикутна матриця, то

$$y_i = \frac{b_i - \sum_{j=1}^{i-1} s_{j,i} d_{j,j} y_j}{s_{i,i} d_{i,i}}, \quad (28)$$

для  $i = \overline{1, n}$  і

$$x_i = \frac{y_i - \sum_{j=1}^{i-1} s_{i,j} x_j}{s_{i,i}}, \quad (29)$$

для  $i = \overline{n-1, 1}$ , де  $x_n = y_n / s_{n,n}$ .

Метод застосовується лише для симетричних матриць. Його складність  $Q = n^3/3 + O(n^2)$ .

**Переваги** цього методу:

- він витрачає в 2 рази менше пам'яті ніж метод Гаусса для зберігання  $A^\top = A$  (необхідний об'єм пам'яті

$$n(n+1)/2 \sim n^2/2;$$

- метод однорідний, без перестановок;
- якщо матриця  $A$  має багато нульових елементів, то і матриця  $S$  також.

Остання властивість дає економію в пам'яті та кількості арифметичних операцій. Наприклад, якщо  $A$  має  $m$  ненульових стрічок по діагоналі ( $m$ -діагональна), то  $Q = O(m^2n)$ .

### 3.3. Обчислення визначника та оберненої матриці

Література:

- Самарский, Гулин, стор. 67–69: [djvu](#), [pdf](#);
- Березин, Жидков, том II, стор. 17–19: [djvu](#), [pdf](#).

Кількість операцій обчислення детермінанту за означенням —  $Q_{\det} = n!$ . В методі Гаусса —  $PA = LU$ . Тому

$$\det P \det A = \det L \det U \tag{30}$$

звідки

$$\det A = (-1)^\ell \det L \det U = (-1)^\ell \prod_{k=1}^n a_{k,k}^{(k)}, \tag{31}$$

де  $\ell$  — кількість перестановок. Ясно, що за методом Гаусса

$$Q_{\det} = \frac{2}{3} \cdot n^3 + O(n^2) \tag{32}$$

В методі квадратного кореня  $A = S^\top D S$ . Тому

$$\det A = \det S^\top \det D \det S = \prod_{k=1}^n d_{k,k} \prod_{k=1}^n s_{k,k}^2. \quad (33)$$

Тепер  $Q_{\det} = n^3/3 + O(n^2)$ .

За означенням

$$AA^{-1} = E, \quad (34)$$

де  $A^{-1}$  обернена до матриці  $A$ . Позначимо

$$A^{-1} = (\alpha_{i,j})_{i,j=1}^n. \quad (35)$$

Тоді  $\vec{\alpha}_j = (\alpha_{i,j})_{i=1}^n$  — вектор-стовпчик оберненої матриці. З (34) маємо

$$A\vec{\alpha}_j = \vec{e}_j, \quad j = \overline{1, n}. \quad (36)$$

де  $\vec{e}_j$  — стовпчики одиничної матриці:  $\vec{e}_j = (\delta_{i,j})_{i=1}^n$ ,

$$\delta_{i,j} = \begin{cases} 1, & i = j, \\ 0, & i \neq j. \end{cases} \quad (37)$$

Для знаходження  $A^{-1}$  необхідно розв'язати  $n$  систем. Для знаходження  $A^{-1}$  методом Гаусса необхідна кількість операцій  $Q = 2n^3 + O(n^2)$ .

### 3.4. Метод прогонки

Література:

- Самарський, Гулин, стор. 45–47: [djvu](#), [pdf](#);

Це економний метод для розв'язання СЛАР з три діагональною матрицею:

$$\begin{cases} -c_0y_0 + b_0y_1 = -f_0, \\ a_iy_{i-1} - c_iy_i + b_iy_{i+1} = -f_i, \\ a_Ny_{N-1} - c_Ny_N = -f_N. \end{cases} \quad (38)$$

Матриця системи

$$A = \begin{pmatrix} -c_0 & b_1 & & 0 \\ a_0 & \ddots & \ddots & \\ & \ddots & \ddots & b_N \\ 0 & & a_N & -c_N \end{pmatrix} \quad (39)$$

тридіагональна.

Розв'язок представимо у вигляді

$$y_i = \alpha_{i+1}y_{i+1} + \beta_{i+1}, \quad i = \overline{0, N-1}. \quad (40)$$

Замінимо в (40) і  $i \mapsto i-1$  і підставимо в (33), тоді

$$(a_i\alpha_i - c_i) \cdot y_i + b_iy_{i+1} = -f_i - a_i\beta_i \quad (41)$$

Звідси

$$y_i = \frac{b_i}{c_i - a_i\alpha_i} \cdot y_{i+1} + \frac{f_i + a_i\beta_i}{c_i - a_i\alpha_i}. \quad (42)$$

Тому з (36)

$$\alpha_{i+1} = \frac{b_i}{c_i - a_i\alpha_i}, \quad \beta_{i+1} = \frac{f_i + a_i\beta_i}{c_i - a_i\alpha_i}, \quad i = \overline{1, N-1}. \quad (43)$$

Умова розв'язності (38) —  $c_i - a_i\alpha_i \neq 0$ .

Щоб знайти всі  $\alpha_i, \beta_i$ , треба задати перші значення. З (38):

$$\alpha_1 = \frac{b_0}{c_0}, \quad \beta_1 = \frac{f_0}{c_0}. \quad (44)$$

Після знаходження всіх  $\alpha_i, \beta_i$  обчислюємо  $y_N$  з системи

$$\begin{cases} a_N y_N - c_N y_N = -f_N, \\ y_{N-1} = \alpha_N y_N + \beta_N. \end{cases} \quad (45)$$

Звідси

$$y_N = \frac{f_N + a_N \beta_N}{c_N - a_N \alpha_N}. \quad (46)$$

**Алгоритм:**

```
alpha[1], beta[1] = b[0] / c[0], f[0] / c[0]

for i in range(1, N):
    z[i] = c[i] - a[i] * alpha[i]
    alpha[i + 1], beta[i + 1] = b[i] / z[i], \
        (f[i] + a[i] * beta[i]) / z[i]

y[N] = (f[N] + a[N] * beta[N]) / \
    (c[N] - a[N] * alpha[N])

for i in range(N - 1, -1, -1):
    y[i] = alpha[i + 1] * y[i + 1] + beta[i + 1]
```

Складність алгоритму  $Q = 8N - 2$ .

Метод можна застосовувати, коли  $c_i - a_i \alpha_i \neq 0, \forall i : |\alpha_i| \leq 1$ . Якщо  $|\alpha_i| \geq q > 1$  то  $\Delta y_0 \geq q^N \Delta y_N$  (тут  $\Delta y_i$  абсолютна похибка обчислення  $y_i$ ), а це приводить до експоненціального накопичення похибок заокруглення, тобто нестійкості алгоритму прогонки.

**Теорема** (про достатні умови стійкості метода прогонки): Нехай  $a_i, b_i \neq 0$ , та

$$|c_i| \geq |a_i| + |b_i|, \quad \forall i, \quad a_0 = b_N = 0, \quad (47)$$

та хоча би одна нерівність строга. Тоді  $|\alpha_i| \leq 1$  та

$$z_i = c_i - a_i \alpha_i \neq 0, \quad i = \overline{1, N}. \quad (48)$$

**Задача 8:** Довести теорему про стійкість методу прогонки.

### 3.5. Обумовленість систем лінійних алгебраїчних рівнянь

Література:

- Самарський, Гулин, стор. 74–81: [djvu](#), [pdf](#);

Нехай задано СЛАР

$$A\vec{x} = \vec{b}. \quad (49)$$

Припустимо, що матриця і права частина системи задані неточно і фактично розв'язуємо систему

$$B\vec{y} = \vec{h}, \quad (50)$$

де  $B = A + C$ ,  $\vec{h} = \vec{b} + \vec{\eta}$ ,  $\vec{y} = \vec{x} + \vec{z}$ .

Малість детермінанту  $\det A \ll 1$  не є необхідною умовою різкого збільшення похибки. Це ілюструє наступний приклад:

$$A = \text{diag}(\varepsilon), \quad a_{i,j} = \varepsilon \delta_{i,j}. \quad (51)$$

Тоді  $\det A = \varepsilon^n \ll 1$ , але  $x_i = b_i/\varepsilon$ . Тому  $\Delta x_i = \Delta b_i/\varepsilon \gg 1$ .

Оцінимо похибку розв'язку. Підставивши значення  $B$ ,  $\vec{h}$ , та  $\vec{z} = \vec{y} - \vec{x}$ , отримаємо:

$$(A + C)(\vec{x} + \vec{z}) = \vec{b} + \vec{\eta}. \quad (52)$$

Віднімемо від цієї рівності (49) у вигляді  $A\vec{z} + C\vec{x} + C\vec{z} = \vec{\eta}$ . Тоді

$$A\vec{z} = \vec{\eta} - C\vec{x} - C\vec{z}, \quad \vec{z} = A^{-1}(\vec{\eta} - C\vec{x} - C\vec{z}). \quad (53)$$

**Означення:** Введемо норми векторів:  $\|\vec{z}\|$ :

$$\|\vec{z}\|_1 = \sum_{i=1}^n |z_i|, \quad (54)$$

$$\|\vec{z}\|_2 = \left( \sum_{i=1}^n |z_i|^2 \right)^{1/2}, \quad (55)$$

$$\|\vec{z}\|_\infty = \max_i |z_i|. \quad (56)$$

**Означення:** Норми матриці, що відповідають нормам вектора, тобто

$$|A|_m = \sup_{\|\vec{x}\|_m \neq 0} \frac{|A\vec{x}|_m}{\|\vec{x}\|_m}, \quad m = 1, 2, \infty. \quad (57)$$

такі:

$$|A|_1 = \max_j \sum_{i=1}^n |a_{i,j}|, \quad (58)$$

$$|A|_2 = \max_i \sqrt{\lambda_i(A^\top A)}, \quad (59)$$

$$|A|_\infty = \max_i \sum_{j=1}^n |a_{i,j}|, \quad (60)$$

де  $\lambda_i(B)$  — власні значення матриці  $B$ .

Позначимо  $\delta(\vec{x}) = \|\vec{z}\|/\|\vec{x}\|$ ,  $\delta(\vec{b}) = \|\vec{\eta}\|/\|\vec{b}\|$ ,  $\delta(A) = \|C\|/\|A\|$  — відносні похибки  $\vec{x}$ ,  $\vec{b}$ ,  $A$ , де  $\|\cdot\|_k$  — одна з введених вище норм.

Для характеристики зв'язку між похибками правої частини та розв'язку вводять поняття обумовленості матриці системи.

**Означення:** Число обумовленості матриці  $A$  —  $\text{cond}(A) = \|A\| \cdot \|A^{-1}\|$ .

**Теорема:** Якщо  $\exists A^{-1}$  та  $\|A^{-1}\| \cdot \|C\| < 1$ , то

$$\delta(\vec{x}) \leq \frac{\text{cond}(A)}{1 - \text{cond}(A) \cdot \delta(A)} (\delta(A) + \delta(\vec{b})), \quad (61)$$

де  $\text{cond}(A)$  — число обумовленості.

*Доведення:*

$$A\vec{z} = \vec{\eta} - C\vec{x} - C\vec{z}, \quad \vec{z} = A^{-1}\vec{\eta} - A^{-1}C\vec{x} - A^{-1}C\vec{z} \quad (62)$$

$$\begin{aligned} |\vec{z}| &\leq |A^{-1}\vec{\eta}| + |A^{-1}C\vec{x}| + |A^{-1}C\vec{z}| \leq \\ &\leq |A^{-1}| \cdot |\vec{\eta}| + |A^{-1}| \cdot |C| \cdot |\vec{x}| + |A^{-1}| \cdot |C| \cdot |\vec{z}|. \end{aligned} \quad (63)$$

$$|\vec{z}| \leq \frac{|A^{-1}| \cdot (|\vec{\eta}| + |C| \cdot |\vec{x}|)}{1 - |A^{-1}| \cdot |C|} \quad (64)$$

Оцінка похибки

$$\begin{aligned} \delta(\vec{x}) &\leq \frac{|A^{-1}|}{1 - |A^{-1}| \cdot |C|} \left( \frac{|\vec{\eta}|}{|\vec{x}|} + |C| \right) = \\ &= \frac{|A^{-1}| \cdot |A|}{1 - |A^{-1}| \cdot |A| \cdot \frac{|C|}{|A|}} \left( \frac{|\vec{\eta}|}{|A| \cdot |\vec{x}|} + \delta(A) \right) \leq \quad (65) \\ &\leq \frac{\text{cond}(A)}{1 - \text{cond}(A) \cdot \delta(A)} \left( \frac{|\vec{\eta}|}{|\vec{x}|} + \delta(A) \right). \quad \square \end{aligned}$$

**Наслідок:** Якщо  $C \equiv 0$ , то  $\delta(\vec{x}) \leq \text{cond}(A) \cdot \delta(\vec{b})$ .

**Властивості  $\text{cond}(A)$ :**

- $\text{cond}(A) \geq 1$ ;
- $\text{cond}(A) \geq \max_i |\lambda_i(A)| / \min_i |\lambda_i(A)|$ ;
- $\text{cond}(AB) \leq \text{cond}(A) \cdot \text{cond}(B)$ ;
- $A^\top = A^{-1} \implies \text{cond}(A) = 1$ .

Друга властивість має місце оскільки довільна норма матриці не менше її найбільшого за модулем власного значення. Значить  $\|A\| \geq \max |\lambda_A|$ . Оскільки власні значення матриць  $A^{-1}$  та  $A$  взаємно обернені, то

$$|A^{-1}| \geq \max \frac{1}{|\lambda_A|} = \frac{1}{\min |\lambda_A|}. \quad (66)$$

Якщо  $1 \ll \text{cond}(A)$ , то система називається *погано обумовленою*.

Оцінка впливу похибок заокруглення при обчисленні розв'язку СЛАР така (Дж. Уілкінсон):  $\delta(A) = O(n\beta^{-t})$ ,  $\delta(\vec{b}) = O(\beta^{-t})$ , де  $\beta$  — розрядність ЕОМ,  $t$  — кількість розрядів, що відводиться під мантису числа. З оцінки (61) витікає:  $\delta(\vec{x}) = \text{cond}(A) \cdot O(n\beta^{-t})$ . Висновок: найпростіший спосіб підвищити точність обчислення розв'язку погано обумовленої СЛАР — збільшити розрядність ЕОМ при обчисленнях. Інші способи пов'язані з розглядом цієї СЛАР як некоректної задачі із застосуванням відповідних методів її розв'язання.

**Приклад** погано обумовленої системи — системи з матрицею Гільберта

$$H_n = \left( \frac{1}{i+j-1} \right)_{i,j=1}^n, \quad (67)$$

наприклад  $\text{cond}(H_8) \approx 10^9$ .

[Назад до лекцій](#)

[Назад на головну](#)

# Numerical Analysis

established about 4'000 years ago

- 4. Ітераційні методи для систем
  - 4.1. Ітераційні методи розв'язання СЛАР
    - 4.1.1. Метод простої ітерації
    - 4.1.2. Метод Якобі
    - 4.1.3. Метод Зейделя
    - 4.1.4. Матрична інтерпретація методів Якобі і Зейделя
    - 4.1.5. Однокрокові (двошарові) ітераційні методи
    - 4.1.6. Збіжності стаціонарних ітераційних процесів у випадку симетричних матриць
    - 4.1.7. Метод верхньої релаксації
    - 4.1.8. Методи варіаційного типу
  - 4.2. Методи розв'язання нелінійних систем
    - 4.2.1. Метод простої ітерації
    - 4.2.2. Метод Ньютона
    - 4.2.3. Модифікований метод Ньютона

## 4. Ітераційні методи для систем

### 4.1. Ітераційні методи розв'язання СЛАР

Література:

- Самарский, Гулин, стор. 82–102: [djvu](#), [pdf](#);
- Березин, Жидков, том II, стор. 54–67: [djvu](#), [pdf](#).

Систему

$$A\vec{x} = \vec{b} \quad (1)$$

зводимо до вигляду

$$\vec{x} = B\vec{x} + \vec{f}. \quad (2)$$

Будь яка система

$$\vec{x} = \vec{x} - C \cdot (A\vec{x} - \vec{b}) \quad (3)$$

має вигляд (2) і при  $\det C \neq 0$  еквівалентна системі (1).

Наприклад, для  $C = \tau \cdot E$ :

$$\vec{x} = \vec{x} - \tau \cdot (A\vec{x} - \vec{b}). \quad (3')$$

#### 4.1.1. Метод простої ітерації

Цей метод застосовується до рівняння (2)

$$\vec{x}^{(k+1)} = B\vec{x}^{(k)} + \vec{f}, \quad (4)$$

де  $\vec{x}^{(0)}$  — початкове наближення, задано.

**Теорема:** Ітераційний процес збігається, тобто

$$\left| \vec{x}^{(k)} - \vec{x} \right| \xrightarrow[k \rightarrow \infty]{} 0, \quad (5)$$

якщо

$$|B| \leq q < 1. \quad (6)$$

При цьому має місце оцінка

$$\left| \vec{x}^{(n)} - \vec{x} \right| \leq \frac{q^n}{1-q} \cdot \left| \vec{x}^{(1)} - \vec{x}^{(0)} \right|. \quad (7)$$

#### 4.1.2. Метод Якобі

Припустимо  $\forall i: a_{i,i} \neq 0$ . Зведемо систему (1) до вигляду

$$x_i = - \sum_{j=1}^{i-1} \frac{a_{i,j}}{a_{i,i}} \cdot x_j - \sum_{j=i+1}^n \frac{a_{i,j}}{a_{i,i}} \cdot x_j + \frac{b_i}{a_{i,i}}, \quad (8)$$

де  $i = \overline{1, n}$ .

Ітераційний процес запишемо у вигляді

$$x_i^{(k+1)} = - \sum_{j=1}^{i-1} \frac{a_{i,j}}{a_{i,i}} \cdot x_j^{(k)} - \sum_{j=i+1}^n \frac{a_{i,j}}{a_{i,i}} \cdot x_j^{(k)} + \frac{b_i}{a_{i,i}}, \quad (9)$$

де  $k = 0, 1, \dots$ , а  $i = \overline{1, n}$ .

**Теорема:** Ітераційний процес збігається до розв'язку, якщо виконується умова

$$\forall i : \sum_{\substack{j=1 \\ j \neq i}}^n |a_{i,j}| \leq |a_{i,i}|. \quad (10)$$

Це умова діагональної переваги матриці  $A$ .

**Теорема:** Якщо ж

$$\forall i : \sum_{j=1}^n |a_{i,j}| \leq q \cdot |a_{i,i}|, \quad 0 \leq q < 1. \quad (11)$$

то має місце оцінка точності:

$$|\vec{x}^{(n)} - \vec{x}| \leq \frac{q^n}{1-q} \cdot |\vec{x}^{(0)} - \vec{x}|. \quad (12)$$

#### 4.1.3. Метод Зейделя

В компонентному вигляді ітераційний метод Зейделя записується так:

$$x_i^{(k+1)} = - \sum_{j=1}^{i-1} \frac{a_{i,j}}{a_{i,i}} \cdot x_j^{(k+1)} - \sum_{j=i+1}^n \frac{a_{i,j}}{a_{i,i}} \cdot x_j^{(k)} + \frac{b_i}{a_{i,i}}, \quad (13)$$

де  $k = 0, 1, \dots$ , а  $i = \overline{1, n}$ .

На відміну від методу Якобі на  $k$ -му-кроці попередні компоненти розв'язку беруться з  $(k+1)$ -ої ітерації.

**Теорема:** Достатня умова збіжності методу Зейделя —  $A^\top = A > 0$ .

#### 4.1.4. Матрична інтерпретація методів Якобі і Зейделя

Подамо матрицю  $A$  у вигляді

$$A = A_1 + D + A_2, \quad (14)$$

де  $A_1$  — нижній трикутник матриці  $A$ ,  $A_2$  — верхній трикутник матриці  $A$ ,  $D$  — її діагональ. Тоді систему (1) запишемо у вигляді

$$D\vec{x} = A_1\vec{x} + A_2\vec{x} + \vec{b}, \quad (15)$$

або

$$\vec{x} = D^{-1}A_1\vec{x} + D^{-1}A_2\vec{x} + D^{-1}\vec{b}, \quad (16)$$

Матричний запис методу Якобі:

$$\vec{x}^{(k+1)} = D^{-1}A_1\vec{x}^{(k)} + D^{-1}A_2\vec{x}^{(k)} + D^{-1}\vec{b}, \quad (17)$$

методу Зейделя:

$$\vec{x}^{(k+1)} = D^{-1}A_1\vec{x}^{(k+1)} + D^{-1}A_2\vec{x}^{(k)} + D^{-1}\vec{b}, \quad (18)$$

**Теорема:** Необхідна і достатня умова збіжності методу Якобі: всі корені рівняння

$$\det(D + \lambda(A_1 + A_2)) = 0 \quad (19)$$

по модулю більше 1.

**Теорема:** Необхідна і достатня умова збіжності методу Зейделя: всі корені рівняння

$$\det(A_1 + D + \lambda A_2) = 0 \quad (20)$$

по модулю більше 1.

#### 4.1.5. Однокрокові (двошарові) ітераційні методи

Канонічною формою однокрокового ітераційного методу розв'язку СЛАР є його запис у вигляді

$$B_k \frac{\vec{x}^{(k+1)} - \vec{x}^{(k)}}{\tau_{k+1}} + A\vec{x}^{(k)} = \vec{b}, \quad (21)$$

Тут  $\{B_k\}$  — послідовність матриць (пере-обумовлюючі матриці), що задають ітераційний метод на кожному кроці;  $\{\tau_{k+1}\}$  — ітераційні параметри.

**Означення:** Якщо  $B_k = E$ , то ітераційний процес називається **явним**

$$\vec{x}^{(k+1)} = \vec{x}^{(k)} - \tau_{k+1} (A\vec{x}^{(k)} + \vec{b}). \quad (22)$$

**Означення:** Якщо  $B_k \neq E$ , то ітераційний процес називається **неявним**

$$B_k \vec{x}^{(k+1)} = F^k. \quad (23)$$

У цьому випадку на кожній ітерації необхідно розв'язувати СЛАР.

**Означення:** Якщо  $\tau_{k+1} \equiv \tau$ ,  $B_k \equiv B$ , то ітераційний процес називається **стаціонарним**; інакше — **нестаціонарним**.

Методами, що розглянуті вище відповідають:

- методу простої ітерації:  $B_k = E$ ,  $\tau_{k+1} = \tau$ ;
- методу Якобі:  $B_k = D$ ,  $\tau_{k+1} = 1$ ;

- методу Зейделя:  $B_k = D + A_1$ ,  $\tau_{k+1} = 1$ .

#### 4.1.6. Збіжності стаціонарних ітераційних процесів у випадку симетричних матриць

Розглянемо випадок симетричних матриць  $A^T = A$  і стаціонарний ітераційний процес  $B_k \equiv E$ ,  $\tau_{k+1} \equiv \tau$ .

Нехай для  $A$  справедливі нерівності

$$\gamma_1 E \leq A \leq \gamma_2 E, \quad \gamma_1, \gamma_2 > 0. \quad (24)$$

Тоді при виборі  $\tau = \tau_0 = \frac{2}{\gamma_1 + \gamma_2}$  ітераційний процес збігається.

Найбільш точним значенням  $\gamma_1$ ,  $\gamma_2$  при яких виконуються обмеження (24) є  $\gamma_1 = \min \lambda_i(A)$ ,  $\gamma_2 = \max \lambda_i(A)$ . Тоді

$$q = q_0 = \frac{\gamma_2 - \gamma_1}{\gamma_2 + \gamma_1} = \frac{1 - \xi}{1 + \xi}, \quad \xi = \frac{\gamma_1}{\gamma_2}. \quad (25)$$

і справедлива оцінка

$$\left| \vec{x}^{(n)} - \vec{x} \right| \leq \frac{q^n}{1 - q} \cdot \left| \vec{x}^{(0)} - \vec{x} \right|. \quad (26)$$

**Зауваження:** аналогічно обчислюється  $q$  і для методу релаксації розв'язання нелінійних рівнянь, де  $\gamma_1 = m = \min |f'(x)|$ ,  $\gamma_2 = M_1 = \max |f'(x)|$ .

Явний метод з багатьма параметрами  $\{\tau_k\}$ :

$$B \equiv E, \quad \tau_k : \min_{\tau} q(\tau), \quad n = n(\epsilon) \rightarrow \min, \quad (27)$$

які обчислюються за допомогою нулів багаточлена Чебишова, називаються ітераційним методом з чебишевським набором

параметрів.

#### 4.1.7. Метод верхньої релаксації

Узагальненням методу Зейделя є метод верхньої релаксації:

$$(D + \omega A_1) \cdot \frac{\vec{x}^{(k+1)} + \vec{x}^{(k)}}{\omega} + A\vec{x}^{(k)} = \vec{b}, \quad (28)$$

де  $D$  — діагональна матриця з елементами  $a_{i,i}$  по діагоналі.  
 $\omega > 0$  — заданий числовий параметр.

Тепер  $B = D + \omega A_1$ ,  $\tau = \omega$ . Якщо  $A^\top = A > 0$ , то метод верхньої релаксації збігається при умові  $0 < \omega < 2$ . Параметр підбирається експериментально з умови мінімальної кількості ітерацій.

#### 4.1.8. Методи варіаційного типу

До цих методів відносяться: метод мінімальних нев'язок, метод мінімальних поправок, метод найшвидшого спуску, метод спряжених градієнтів. Вони дозволяють обчислювати наближення без використання апріорної інформації про  $\gamma_1$ ,  $\gamma_2$  в (24).

Нехай  $B = E$ . Для методу мінімальних нев'язок параметри  $\tau_{k+1}$  обчислюються з умови

$$\begin{aligned} |\vec{r}^{(k+1)}|^2 &= |\vec{r}^{(k)}|^2 - 2\tau_{k+1} \cdot \langle \vec{r}^{(k)}, A\vec{r}^{(k)} \rangle + \\ &+ \tau_{k+1}^2 \cdot |A\vec{r}^{(k)}|^2 \rightarrow \min. \end{aligned} \quad (29)$$

Тому

$$\tau_{k+1} = \frac{\left\langle A\vec{r}^{(k)}, \vec{r}^{(k)} \right\rangle}{\left| \vec{r}^{(k)} \right|^2}, \quad (30)$$

де  $\vec{r}^{(k)} = A\vec{x}^{(k)} - \vec{b}$  — нев'язка.

Умова для завершення ітераційного процесу:

$$\left| \vec{r}^{(n)} \right| < \epsilon. \quad (31)$$

Швидкість збіжності цього методу співпадає із швидкістю методу простої ітерації з одним оптимальним параметром  $\tau_0 = \frac{2}{\gamma_1 + \gamma_2}$ .

Аналогічно будуються методи з  $B \neq E$ . Матриця  $B$  називається переобумовлювачем і дозволяє підвищити швидкість збіжності ітераційного процесу. Його вибирають з умов

- легко розв'язувати СЛАР  $B\vec{x}^{(k)} = F_k$  (діагональний, трикутній, добуток трикутних та інше);
- зменшення числа обумовленості матриці  $B^{-1}A$  у порівнянні з  $A$ .

## 4.2. Методи розв'язання нелінійних систем

Література:

- Самарский, Гулин, стор. 209–213: [djvu](#), [pdf](#);

Розглянемо систему рівнянь

$$\begin{cases} f_1(x_1, \dots, x_n) = 0, \\ \dots \\ f_n(x_1, \dots, x_n) = 0. \end{cases} \quad (32)$$

Перепишемо її у векторному вигляді:

$$\vec{f}(\vec{x}) = 0. \quad (33)$$

#### 4.2.1. Метод простої ітерації

В цьому методі рівняння (33) зводиться до еквівалентного вигляду

$$\vec{x} = \vec{\Phi}(\vec{x}). \quad (34)$$

Ітераційний процес представимо у вигляді:

$$\vec{x}^{(k+1)} = \vec{\Phi}\left(\vec{x}^{(k)}\right). \quad (35)$$

початкове наближення  $\vec{x}^{(0)}$  — задано.

Нехай оператор  $\vec{\Phi}$  визначений на множині  $H$ . За теоремою про стискуючі відображення ітераційний процес (35) сходиться, якщо виконується умова

$$|\vec{\Phi}(\vec{x}) - \vec{\Phi}(\vec{y})| \leq q \cdot |\vec{x} - \vec{y}|, \quad 0 < q < 1, \quad (36)$$

або

$$|\vec{\Phi}'(\vec{x})| \leq q < 1, \quad (37)$$

де  $\vec{x} \in U_r$ ,  $\vec{\Phi}'(\vec{x}) = \left( \frac{\partial \varphi_i}{\partial x_j} \right)_{i,j=1}^n$ . Для похибки справедлива оцінка

$$|\vec{x}^{(m)} - \vec{x}| \leq \frac{q^n}{1-q} \cdot |\vec{x}^{(0)} - \vec{x}|. \quad (38)$$

Частинним випадком методу простої ітерації є метод релаксації для рівняння (33):

$$\vec{x}^{(k+1)} = \vec{x}^{(k)} - \tau \cdot \vec{F}(\vec{x}^{(k)}), \quad (39)$$

де  $\tau < 2 / \|\vec{F}'(\vec{x})\|$ .

#### 4.2.2. Метод Ньютона

Розглянемо рівняння

$$\vec{F}(\vec{x}) = 0. \quad (40)$$

Представимо його у вигляді

$$\vec{F}(\vec{x}^{(k)}) + \vec{F}'(\vec{\xi}^{(k)}) \cdot (\vec{x} - \vec{x}^{(k)}) = 0, \quad (41)$$

де

$$\vec{\xi}^{(k)} = \vec{x}^{(k)} + \theta_k \cdot (\vec{x}^{(k)} - \vec{x}), \quad (42)$$

де  $0 < \theta_k < 1$ . Тут  $\vec{F}'(\vec{x}) = \left( \frac{\partial f_i}{\partial x_j} \right)_{i,j=1}^n$  — матриця Якобі для  $\vec{F}(\vec{x})$ . Можемо наблизено вважати  $\vec{\xi}^{(k)} \approx \vec{x}^{(k)}$ . Тоді з (41)

матимемо

$$\vec{F} \left( \vec{x}^{(k)} \right) + \vec{F}' \left( \vec{x}^{(k)} \right) \cdot \left( \vec{x}^{(k+1)} - \vec{x}^{(k)} \right) = 0. \quad (43)$$

Ітераційний процес представимо у вигляді:

$$\vec{x}^{(k+1)} = \vec{x}^{(k)} - \vec{F}' \left( \vec{x}^{(k)} \right)^{-1} \cdot \vec{F} \left( \vec{x}^{(k)} \right). \quad (44)$$

Для реалізації методу Ньютона потрібно, щоб існувала обернена матриця

$$\vec{F}' \left( \vec{x}^{(k)} \right)^{-1}. \quad (45)$$

Можна не шукати обернену матрицю, а розв'язувати на кожній ітерації СЛАР

$$A_k \vec{z}^{(k)} = \vec{F} \left( \vec{x}^{(k)} \right), \quad (46)$$
$$\vec{x}^{(k+1)} = \vec{x}^{(k)} - \vec{z}^{(k)},$$

де  $k = 0, 1, 2, \dots$ , і  $\vec{x}^{(0)}$  — задано, а матриця  $A_k = \vec{F}' \left( \vec{x}^{(k)} \right)$ .

Метод має квадратичну збіжність, якщо добре вибрано початкове наближення. Складність методу (при умові використання методу Гаусса розв'язання СЛАР (46) на кожній ітерації  $Q_n = \frac{2}{3}n^3 + O(n^2)$ , де  $n$  — розмірність системи (33).

#### 4.2.3. Модифікований метод Ньютона

Ітераційний процес має вигляд:

$$\vec{x}^{(k+1)} = \vec{x}^{(k)} - \vec{F}' \left( \vec{x}^{(0)} \right)^{-1} \cdot \vec{F} \left( \vec{x}^{(k)} \right). \quad (47)$$

Тепер обернена матриця обчислюється тільки на нульовій ітерації. На інших — обчислення нового наближення зводиться до множення матриці  $A_0 = \vec{F}' \left( \vec{x}^{(0)} \right)^{-1}$  на вектор  $\vec{F} \left( \vec{x}^{(k)} \right)$  та додавання до  $\vec{x}^{(k)}$ .

Запишемо метод у вигляді системи лінійних рівнянь (аналог (46))

$$\begin{aligned} A_0 \vec{z}^{(k)} &= \vec{F} \left( \vec{x}^{(k)} \right), \\ \vec{x}^{(k+1)} &= \vec{x}^{(k)} - \vec{z}^{(k)}, \end{aligned} \quad (48)$$

де  $k = 0, 1, 2, \dots$

Оскільки матриця  $A_0$  розкладається на трикутні (або обертається) один раз, то складність цього методу на одній ітерації (окрім нульової)  $Q_n = O(n^2)$ . Але цей метод має лінійну швидкість збіжності.

Можливе циклічне застосування модифікованого методу Ньютона, тобто коли обернену матрицю похідних шукаємо та обертаємо через певне число кроків ітераційного процесу.

**Задача 9:** Побудувати аналог методу січних для систем нелінійних рівнянь.

[Назад до лекцій](#)

[Назад на головну](#)

---

**numerical-analysis** is maintained by [csc-knu](#).

© 2019 Київський національний університет імені Тараса Шевченка, Андрій Риженко, Скибицький Нікіта

# Numerical Analysis

established about 4'000 years ago

- 5. Алгебраїчна проблема власних значень
  - 5.1. Степеневий метод
  - 5.2. Ітераційний метод обертання

## 5. Алгебраїчна проблема власних значень

Нехай задано матрицю  $A \in \mathbb{R}^{n \times n}$ . Тоді задача на власні значення ставиться так: знайти число  $\lambda$  та вектор  $x \neq 0$ , що задовольняють рівнянню

$$Ax - \lambda x. \quad (1)$$

**Означення:**  $\lambda$  називається власним значенням  $A$ , а  $x$  — власним вектором.

З (1)

$$\begin{aligned} \det(A - \lambda E) &= P_n(\lambda) = \\ &= (-1)^n \lambda^n + a_n \lambda^{n-1} + \dots + a_0 = 0. \end{aligned} \quad (2)$$

Тут  $P_n(\lambda)$  — характеристичний багаточлен.

Для розв'язання багатьох задач механіки, фізики, хімії потрібне знаходження всіх власних значень  $\lambda_i, i = \overline{1, n}$ , а іноді й всіх

власних векторів  $x_i$ , що відповідають  $\lambda_i$ .

**Означення:** Цю задачу називають *повною проблемою власних значень*.

В багатьох випадках потрібно знайти лише максимальне або мінімальне за модулем власне значення матриці. При дослідженні стійкості коливальних процесів іноді потрібно знайти два максимальних за модулем власних значення матриці.

**Означення:** Останні дві задачі називають *частковими проблемами власних значень*.

## 5.1. Степеневий метод

Література:

- БЖК, стор. 309–314;
- КБМ, стор. 149–157.

1. Знаходження  $\lambda_{\max}$ :  $\lambda_{\max} \equiv |\lambda_1| \geq |\lambda_2| \geq |\lambda_3| \geq \dots$

Нехай  $x^{(0)}$  — заданий вектор, будемо послідовно обчислювати вектори

$$x^{(k+1)} = Ax^{(k)}, \quad k = 0, 1, \dots \quad (3)$$

Тоді  $x^{(k)} = A^k x^{(0)}$ . Нехай  $\{e_i\}_{i=1}^n$  — система власних векторів. Представимо  $x^{(0)}$  у вигляді:

$$x^{(0)} = \sum_{i=1}^n c_i e_i. \quad (4)$$

Оскільки  $Ae_i = \lambda_i e_i$ , то  $x^{(k)} = \sum_{i=1}^n c_i \lambda_i^k e_i$ . При великах  $k$ :  
 $x^{(k)} \approx c_1 \lambda_1^k e_1$ . Тому

$$\mu_1^{(k)} = \frac{x_m^{(k+1)}}{x_m^{(k)}} = \lambda_1 + O\left(\left|\frac{\lambda_2}{\lambda_1}\right|^k\right). \quad (5)$$

Значить  $\mu_1^{(k)} \xrightarrow[k \rightarrow \infty]{} \lambda_1$ .

Якщо матриця  $A = A^\top$  симетрична, то існує ортонормована система векторів  $\langle e_i, e_j \rangle = \delta_{ij}$ . Тому

$$\begin{aligned}
\mu_1^{(k)} &= \frac{\langle x^{(k+1)}, x^{(k)} \rangle}{\langle x^{(k)}, x^{(k)} \rangle} = \\
&= \frac{\left\langle \sum_i c_i \lambda_i^{k+1} e_i, \sum_j c_j \lambda_j^k e_j \right\rangle}{\left\langle \sum_i c_i \lambda_i^k e_i, \sum_j c_j \lambda_j^k e_j \right\rangle} = \\
&= \frac{\sum_i c_i^2 \lambda_i^{2k+1}}{\sum_i c_i^2 \lambda_i^{2k}} = \\
&= \frac{c_1^2 \lambda_1^{2k+1} + c_2^2 \lambda_2^{2k+1} + \dots}{c_1^2 \lambda_1^{2k} + c_2^2 \lambda_2^{2k} + \dots} = \\
&= \lambda_1 + O\left(\left|\frac{\lambda_2}{\lambda_1}\right|^{2k}\right) \xrightarrow{k \rightarrow \infty} \lambda_1.
\end{aligned} \tag{6}$$

Це означає збіжність до максимального за модулем власного значення з квадратичною швидкістю.

Якщо  $|\lambda_1| > 1$ , то при проведенні ітерацій відбувається зрост компонент вектора  $x^{(k)}$ , що приводить до «переповнення» (overflow). Якщо ж  $|\lambda_1| < 1$ , то це приводить до зменшення компонент (underflow). Позбутися негативу такого явища можна нормуючи вектори  $x^{(k)}$  на кожній ітерації.

Алгоритм степеневого методу знаходження максимального за модулем власного значення з точністю  $\varepsilon$  виглядає так:

```

e[0] = x[0] / norm(x[0])

k = 0
while True:
    k += 1

    x[k + 1] = A * x[k]
    mu[k][1] = scalar_product(x[k + 1], e[k])
    e[k + 1] = x[k + 1] / norm(x[k + 1])

    if abs(mu[k + 1][1] - mu[k][1]) < epsilon:
        break

lambda_1 = mu[k + 1][1]

```

За цим алгоритмом для симетричної матриці  $A^T = A$  швидкість прямування  $\mu_1^{(k)}$  до  $\lambda_{\max}$  — квадратична.

2. Знаходження  $\lambda_2$ :  $|\lambda_1| \geq |\lambda_2| \geq |\lambda_3| \geq \dots$  Нехай  $\lambda_1, e_1$  відомі.

**Задача 10:** Довести, що якщо  $|\lambda_1| \geq |\lambda_2| \geq |\lambda_3| \geq \dots$  то

$$\mu_2^{(k)} = \frac{x_m^{(k+1)} - \lambda_1 x_m^{(k)}}{x_m^{(k)} - \lambda_1 x_m^{(k-1)}} \xrightarrow{k \rightarrow \infty} \lambda_2, \quad (7)$$

де  $x^{(k+1)} = Ax^{(k)}$ ,  $x_m^{(k)}$  —  $m$ -та компонента  $x^{(k)}$ .

**Задача 11:** Побудувати алгоритм обчислення  $\lambda_2, e_2$ , використовуючи нормування векторів та скалярні добутки для обчислення  $\mu_2^{(k)}$ .

3. Знаходження мінімального власного числа

$$\lambda_{\min}(A) = \min_i |\lambda_i(A)|.$$

Припустимо, що  $\lambda_i(a) > 0$  то відоме  $\lambda_{\max}$ . Розглянемо матрицю  $B = \lambda_{\max}E - A$ . Маємо

$$\forall i : \quad \lambda_i(B) = \lambda_{\max} - \lambda_i(A). \quad (8)$$

Тому  $\lambda_{\max}(B) = \lambda_{\max}(A) - \lambda_{\min}(A)$ . Звідси  $\lambda_{\min}(A) = \lambda_{\max}(A) - \lambda_{\max}(B)$ .

Якщо  $\exists i: \lambda_i(A) < 0$ , то будуємо матрицю  $\bar{A} = \sigma E + A$ ,  $\sigma > 0: \bar{A} > 0$  і для неї попередній розгляд дає необхідний результат. Замість  $\lambda_{\max}$  в матриці  $B$  можна використовувати  $\|A\|$ .

Ще один спосіб обчислення мінімального власного значення полягає в використання обернених ітерацій:

$$Ax^{(k+1)} = x^{(k)}, \quad k = 0, 1, \dots \quad (9)$$

Але цей метод вимагає більшої кількості арифметичних операцій: складність методу на основі формули (3):

$Q = O(n^2)$ , а на основі (9) —  $Q = O(n^3)$ , оскільки треба розв'язувати СЛАР, але збігається метод (9) швидше.

## 5.2. Ітераційний метод обертання

Література:

- КБМ, стор. 157–161.

Це метод розв'язання повної проблеми власних значень для симетричних матриць  $A^\top = A$ . Існує матриця  $U$ , що приводить матрицю  $A$  до діагонального виду:

$$A = U\Lambda U^\top, \quad (10)$$

де  $\Lambda$  — діагональна матриця, по діагоналі якої стоять власні значення  $\lambda_i$ ;  $U$  — унітарна матриця, тобто:  $U^{-1} = U^\top$ .

З (1) маємо

$$\Lambda = U^\top AU. \quad (11)$$

Нехай  $\exists \tilde{U}$  — матриця, така що  $\tilde{\Lambda} = \tilde{U}^\top A \tilde{U}$  і  $\tilde{\Lambda} = (\tilde{\lambda}_{ij})_{i,j=1}^n$ ,  
 $|\tilde{\lambda}_{ij}| < \delta \ll 1, i \neq j$ .

Тоді діагональні елементи мало відрізняються від власних значень

$$|\tilde{\lambda}_{ij} - \lambda_i(A)| < \varepsilon = \varepsilon(\delta). \quad (12)$$

Введемо

$$t(A) = \sum_{\substack{i,j=1 \\ i \neq j}}^n a_{ij}^2. \quad (13)$$

З малості величини  $t(A)$  випливає, що діагональні елементи малі. По  $A = A_0$  за допомогою матриць обертання  $U_k$ :

$$U_k = \begin{pmatrix} 1 & \cdots & 0 & \cdots & 0 & \cdots & 0 \\ \vdots & \ddots & \vdots & \ddots & \vdots & \ddots & \vdots \\ 0 & \cdots & \cos \phi & \cdots & -\sin \phi & \cdots & 0 \\ \vdots & \ddots & \vdots & \ddots & \vdots & \ddots & \vdots \\ 0 & \cdots & \sin \phi & \cdots & \cos \phi & \cdots & 0 \\ \vdots & \ddots & \vdots & \ddots & \vdots & \ddots & \vdots \\ 0 & \cdots & 0 & \cdots & 0 & \cdots & 1 \end{pmatrix}. \quad (14)$$

що повертають систему векторів на кут  $\varphi$ , побудуємо послідовність  $\{A_k\}$  таку, що  $A_k \xrightarrow[k \rightarrow \infty]{} \Lambda$ .

**Задача 12:** Показати, що матриця обертання  $U_k$  є унітарною:  $U_k^{-1} = U_k^T$ .

Послідовно будуємо:

$$A_{k+1} = U_K^T A_k U_k, \quad (15)$$

**Означення:** Процес (15) називається **монотонним**, якщо:  $t(A_{k+1}) < t(A_k)$ .

**Задача 13:** Довести, що для матриці (15) виконується:

$$a_{i,j}^{(k+1)} = a_{i,j}^{(k)} \cos(2\varphi) + \frac{1}{2} \left( a_{j,j}^{(k)} - a_{i,i}^{(k)} \right) \sin(2\varphi), \quad (16)$$

Показати, що

$$t(A_{k+1}) = t(A_k) - 2 \left( a_{i,j}^{(k)} \right)^2 \quad (17)$$

якщо вибирати  $\varphi$  з умови  $a_{i,j}^{(k+1)} = 0$ .

Звідси

$$\varphi = \varphi_k = \frac{1}{2} \arctan \left( p^{(k)} \right), \quad (18)$$

тобто

$$p^{(k)} = \frac{2a_{i,j}^{(k)}}{a_{i,i}^{(k)} - a_{j,j}^{(k)}}, \quad (19)$$

де

$$\left| a_{i,j}^{(k)} \right| = \max_{\substack{m,l \\ m \neq l}} \left| a_{m,l}^{(k)} \right|. \quad (20)$$

Тоді  $t(A_k) \xrightarrow{k \rightarrow \infty} 0$ . Чим більше  $n$  тим більше ітерацій необхідно для зведення  $A$  до  $\Lambda$ .

Якщо матриця несиметрична, то застосовують QR- або QL- методи.

[Назад до лекцій](#)

[Назад на головну](#)

---

**numerical-analysis** is maintained by **csc-knu**.

© 2019 Київський національний університет імені Тараса Шевченка, Андрій Риженко, Скибицький Нікіта