ЗМІСТ

ВСТУП

Задачею обчислювальної математики є одержання наближених чисельних рішень, коли це неможливо зробити аналітичними методами, або ж коли використання аналітичних методів є надто складним та трудомістким. Як правило задачі, методи розв’язку яких будуть розглядатися далі, виникають при роботі з математичними моделями.

Математична модель - це спрощене описання реальності за допомогою математичних понять . Математичне моделювання – процес побудови математичних моделей реальних процесів і явищ, тобто метод дослідження об’єктів і процесів реального світу за допомогою наближених описань на мові математичних відношень. В останній час ефективним засобом при роботі з математичними моделями став обчислювальний експеримент.

Обчислювальний експеримент – це інформаційна технологія призначена для вивчення явищ навколишнього світу, коли натурний експеримент є неможливим (наприклад, при вивченні здоров’я людини), або надто небезпечним(наприклад, при вивченні екологічних явищ), чи надто дорогим і складним (наприклад, при вивченні астрофізичних явищ). До переваг обчислювального експерименту, як методу дослідження, слід віднести що він, як правило дешевше натурного. В цей експеримент легко і безпечно втручатися. Його можна повторювати необхідну кількість разів і перервати в будь який момент. В ході експерименту можна змоделювати умови, які неможливо створити в умовах лабораторії.

На відміну від натурних досліджень він дозволяє накопичувати результати отримані при вивченні будь якого круга задач, а потім ефективно застосовувати їх для розв’язку задач в інших областях. Наприклад, рівняння нелінійної теплопровідності описує не тільки теплові процеси, але і дифузії речовини, рух ґрунтових від , фільтрації газу в пористих середовищах, міняється тільки фізичний зміст величин, які входять . в це рівняння.

При рішенні складних задач звичайно доводиться застосовувати декілька методів. Так на початкових (допоміжних) етапах це може бути чисельне диференціювання, інтегрування, визначення власних значень матриць, а на заключному розв’язок системи лінійних алгебраїчних рівнянь(СЛАР). Тому ефективність рішення задачі в цілому повністю визначається характеристиками методів, що використовуються для кожної складової.

РОЗДІЛ 1 ОСНОВНІ ПОНЯТТЯ

1.1 Похибки обчислень.

Класифікація похибок. При чисельному розв'язку математичних і прикладних задач на тому або іншому етапі майже неминуче виникають похибки. Похибкою називається відхилення наближеного розв'язку від істинного розв'язку. Розрізняють наступні типи похибок.

1) **Похибка, яка не усувається .** Вона пов'язана з наближеним характером початкової змістової моделі, а також її математичного описання, зокрема, неможливо врахувати всі фактори в процесі вивчення явища, що моделюється. Вона також пов'язана з тим, що параметрами математичної моделі служать звичайно наближені величини, так як неможливо виконати абсолютно точні виміри. Для обчислювача похибку математичної моделі слід вважати такою, яка не усувається(безумовною), хоча постановник задачі іноді може її змінити.

2) **Похибка методу.** Це похибка, яка пов'язана з способом розв'язку поставленої математичної задачі і яка з'являється в результаті заміни початкової математичної моделі на іншу або скінченну послідовність других, наприклад лінійних моделей. При створенні чисельних методів закладається можливість прослідкувати такі похибки і доведення їх до як завгодно малого рівня. Звідси природнє відношення до похибок метода, як до тих, що можна усунути(або умовних).

3) **Обчислювальна похибка** (похибка дій). Цей тип похибок пов'язаний з необхідністю виконувати арифметичні операції над числами усіченими до кількості розрядів, яка залежить від застосованої обчислювальної техніки(якщо,звичайно не використовуються спеціальні програмні засоби, наприклад, арифметика раціональних чисел), тобто обчислювальна похибка обумовлена округленнями.

Всі три описаних типи похибок в сумі дають **повну похибку** результату розв'язку задачі . Оскільки перший тип похибок не знаходиться в компетенції обчислювача, то для обчислювача він служить лише орієнтиром точності, з якою слід обчислювати математичну модель. Немає рації розв'язувати задачу суттєво точніше, ніж це диктується невизначеністю початкових даних. Таким чином похибку метода підпорядковують похибці задачі. При виводі оцінок похибок чисельних методів звичайно виходять з допущення, що всі операції над числами виконуються точно, це значить що похибки округлення не повинні суттєво відображатися на результатах реалізації методів,тобто повинна бути підпорядкована похибці метода. Вплив похибок округлення не слід випускати з поля зору ні на стадії відбору і алгоритмізації чисельних методів, ні при виборі обчислювальних і програмних засобів, ні при виконанні окремих дій і обчисленні значень функцій.

Так у відомому з фізики рівнянні коливань матеріальної точки

$$m\frac{d^2x}{dt^2}+\beta\frac{dx}{dt}+cx=0, \qquad (1.1)$$

де m – маса точки, β – коефіцієнт опору, а c – жорсткість пружини, параметри $m\ ,\beta,c$ визначаються шляхом вимірювання і мають похибку обумовлену точністю вимірювальних приладів. Якщо врахувати, що сила опору і жорсткість пружини є нелінійними функціями відповідно швидкості і зміщення то маємо рівняння

$$m\frac{d^2x}{dt^2}+\beta_1\frac{dx}{dt}+\beta_2\left(\frac{dx}{dt}\right)^2+\beta_3\left(\frac{dx}{dt}\right)^3+\ldots+c_1x+c_2x^2+c_3x^3+\ldots=0, \quad (1.2)$$

яке більш точно описує процес коливань. Похибка метода виникає тому, що чисельними методом рішається друга, більш проста задача, яка є наближенням вихідної. Так рівняння (1) є наближенням рівняння (2). Процес заміни нелінійної задачі лінійною називається лінеаризацією, тобто (1) одержано шляхом лінеаризації (2).

Приклад. Нехай треба обчислити площу фігури, що обмежена деякою кривою $y = f(x)$, відрізками прямих $x = a, b$ і віссю абсцис. Нехай S - істинне значення площі даної фігури. В якості математичної моделі для обчислення площі візьмемо інтеграл $\int_a^b f(x)dx$. Неточність у цьому виразі закладена в числах a і b а також в функції $y = f(x)$, яка апроксимує

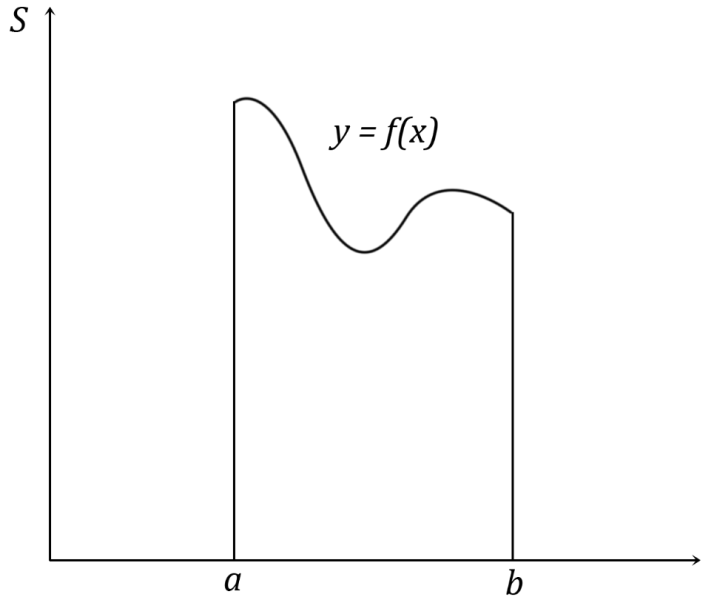


Рис. 1.1.- Модель обчислення площі

криволінійну границю області. Нехай $S_1 = \int_{a_i}^b f(x)dx$ - істинне значення *інтегралу. Тоді різниця $S - S_1$ і буде похибкою, яку неможливо усунути. Для обчислення інтегралу застосуємо чисельний метод, допустимо інтегральну сумі $\sum_{i=1}^{N} \omega_i f(x_i)$. Нехай S_2 - істинне значення суми. Різниця $S_1 - S_2$ - це похибка метода . Нехай, на кінець, S_3 - той результат, який отримаємо після обчислення суми. З за округлень отримаємо величину, яка взагалі , відрізняється від S_2. Таким чином $S_2 - S_3$ це обчислювальна похибка . Сума трьох вказаних похибок, яка дорівнює $S - S_3$, тобто різниці між істинним значенням площі(нам невідомим) і тим значенням, яке ми приймаємо в якості розв'язку задачі, і є повною похибкою.

Деякі алгоритми є досить чутливими до похибок округлення. При

розрахункам по таким алгоритмам невелика похибка, допущена на будь якому його кроці, може сильно зростати, що в результаті приводить до значної обчислювальної похибки. Такі алгоритми називаються **нестійкими.**

Мабуть самим поширеним видом похибки метода є *похибка усічення*, яка виникає в наслідок наближеної заміна об'єкту, що має нескінченне число вимірів, об'єктом з скінченним числом вимірів. Так при пошуку розв'язку у формі розкладення по базису $f(x)=\sum_{i=0}^{\infty}\alpha_i\varphi(x)$ технічно неможливо взяти верхню границю сумування рівною нескінченності і приходиться заміняти якимось скінченним числом $f(x)=\sum_{i=0}^{N}\alpha_i\varphi(x)$. Задача заходженні функції $f(x)$, визначеної у кожній точці відрізка $x\in[a;b]$, як правило зводиться до обчислення значень цієї функції $f(x_i)$ у дискретній послідовності точок.

$$x_i=a+ih \quad h=\frac{b-a}{N}, \quad i=0,1,\ldots,N$$

На перший погляд здається , що чим детальніше буде проведено розбиття відрізка, тим похибка результату буде менша. Але це справедливо тільки для похибки метода. Зі збільшенням точок зростає обсяг обчислень і значить і похибка округлення. Залежність похибки дискретизації (крива I), похибки округлення (крива II) і повної похибки (крива III) схематично показано на рис.2.

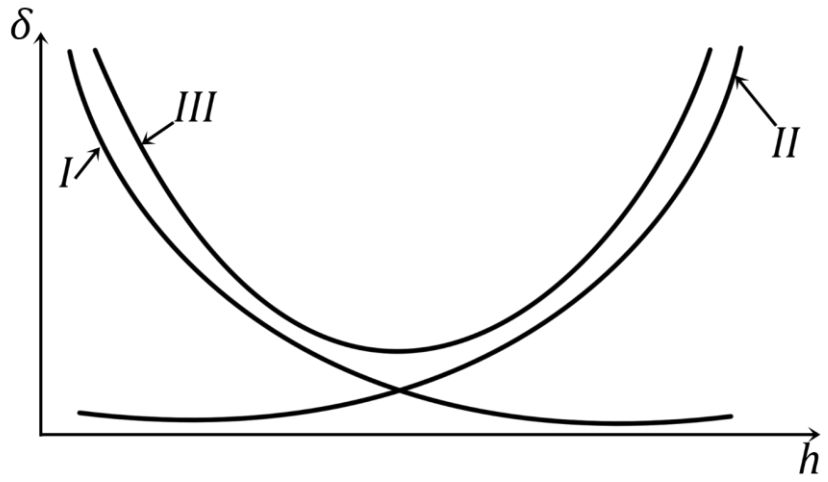


Рис. 1.2.- Залежність похибок від кроку дискретизації

Абсолютна і відносна похибка числа. Коли кажуть, що $\tilde{a}$ являє собою наближене число, то мається на увазі; що існує деяке «точне число » a, яке звичайно нам буває невідоме. Строго кажучи, завдання наближеного числа передбачає завдання двох величин: основного числа $\tilde{a}$ і деякої допоміжної додатного числа $\Delta\tilde{a}$.

Наближенням $\tilde{a}$ числа a називається число, яке мало відрізняється від a і змінює його в обчисленнях. Точність наближених значень характеризують похибкою. Відрізняють два види похибки: абсолютну і відносну. Абсолютною похибкою наближення $\tilde{a}$ називається величина $\Delta\tilde{a}$, яка задовольняє умові $|a-\tilde{a}| \le \Delta\tilde{a}$. Абсолютна похибка пов'язана з розмірністю і не повністю характеризує результат, тому вводиться поняття відносної похибки. Відносною похибкою наближення $\tilde{a}$ називається величина $\delta\tilde{a}$,. яка задовольняє умові $\left|\frac{a-\tilde{a}}{\tilde{a}}\right| \le \delta\tilde{a}$. Далі для відносної похибки приймемо більш зручне позначення ε.

Використання відносної похибки є більш поширеним, так як вона дозволяє проводити аналіз точності метода коли одержані за його допомогою величини мають різний порядок. Так, наприклад, якщо відомі результати двох обчислень, у яких

$$a_1 = 1\cdot 10^{-3},\ a_1^{*} = 2\cdot 10^{-3},\quad \Delta a_1 = 1\cdot 10^{-3},$$

$$a_2 = 2 \cdot 10^6, \ a_2^* = 2{,}1 \cdot 10^6, \qquad \Delta a_2 = 1 \cdot 10^6 ,$$

то на перший погляд, порівнявши тільки абсолютні похибки, можна прийти до висновку, що у першому випадку досягнута вища точність. Але аналіз за допомогою відносної похибки приводить до протилежного висновку.

$\varepsilon_1 = 1 = 100\%, \ \varepsilon_1 = 0{,}1 = 10\% , \ \varepsilon_1 >> \varepsilon_2 .$

Звичайно від методів потребують виконання наступних вимог точність, стійкість, економічність та універсальність.

Якщо похибка метода є його індивідуальною характеристикою, то похибка округлення за своєю природою визначається машинною формою зображення числа. Так у системі з плаваючою крапкою число зображується у вигляді

$$x = \pm q^p \sum_{k=1}^{t} \alpha_k q^{-k}$$

$$x = \pm q^p \sum_{k=1}^{t} \alpha_k q^{-k}$$

тут q - ціле-основа системи числення; $\alpha_1,..,\alpha_t$ - цілі в межах $0 \le \alpha_k \le q$. Довжина мантиси t визначається числом розрядів і є величиною обмеженою $t < t_0$. Очевидно, що при виконанні арифметичних дій, довжина мантиси результату буде більше ніж t_0, і її частина починаючи з α_{t+1} повинна бути відкинутою. Граничне значення t_0 визначається як типом ЕОМ, так і вимогами до системи зображення числа : звичайна точність, подвійна і т.д.

Розглянемо один випадок прояву обчислювальної похибки. Якщо при розв'язанні рівняння

$$x^2 - 140x + 1 = 0$$

Обчислення проводити в десятинній системі числення, при цьому в мантисі числа після округлення отримувати чотири розряду, то

$$x = 70 - \sqrt{4899} \qquad \sqrt{4899} = 69.9992...$$

і після округлення маємо

$$x = 70 - 69{,}99 = 0{,}01$$

Якщо те саме значення $\boldsymbol{x}$ знайти позбувшись ірраціональності,

$$x = \frac{1}{70 + \sqrt{4899}} = \frac{1}{70 + 69{,}99} = \frac{1}{140} = 0.07143.$$

Провівши обчислювання з додатковими розряд можна, перевірити, що в обох випадках всі підкресленні цифри вірні, але в другому випадку точність результату суттєво вище. Справа у тому що у першому випадку прийшлось віднімати два близькі великі числа. Цей ефект має назву втрати значущих цифр при відмінні.

Вважається, що метод є стійким, якщо невеликі зміни вхідних параметрів приводять до невеликих змін в результатах. Економічність методу визначається обсягом обчислень необхідним для його реалізації. На практиці пріоритетним є вимоги точності та стійкості. Якщо виникає потреба вибору одного з кількох методів, то при виконанні цих двох умов проводиться порівняння за іншими характеристиками.

Чисельні методи діляться на два класи прямі та ітераційні. При застосуванні прямих методів алгоритм рішення використовується один раз. Рішення задачі ітераційним методом (методом послідовних наближень) складається з наступних етапів:

- вибір початкового наближення;
- визначення нового наближення;
- перевірка виконання умови зупинки алгоритму;
- перехід до наступного наближення, якщо умова зупинки не виконується.

Умовою зупинки може бути, як величина різниці між двома наближеннями, так і кількість ітерацій. Ітераційні методи застосовуються у тих випадках коли використати прямі методи неможливо, або ж коли відоме початкове наближення близьке до точного. При використанні ітераційних методів суттєвим є поняття збіжності. Якщо $U^{(n)} \to \mathrm{U}$, при $n \to \infty$ де n - номер

наближення, а U - точне рішення, то метод збігається. Практичною мірою збіжності алгоритму є виконання умови

$$\left\| U^{(n+1)} - U^{(n)} \right\| \le \left\| U^{(n)} - U^{(n-1)} \right\|$$

Величина норми обчислюється наступним чином:

для числа $\|U\| = |U|$;

для вектора $\|U\| = \left(\sum_{i=1}^{n} u_i^2 \right)^{1/2}$;

для функції $\|U(x)\| = \left[\int_a^b U^2(x) dx \right]^{1/2}$.

Чисельний метод вважається вдало вибраним, якщо його похибка в декілька разів менше, похибки, що не усувається, а обчислювальна похибка в декілька разів менша похибки метода.

РОЗДІЛ 2 НАБЛИЖЕННЯ ФУНКЦІЙ

2.1 Постановка задачі

Ця задача виникає у двох випадках. У першому, коли для функції даної при дискретних значеннях аргументу у вигляді таблиці (ці значення називаються вузлами інтерполяції) необхідно знайти значення функції у проміжних точках. У другому необхідно деяку «складну функцію» наближено замінити «більш простою». Так операції диференціювання, інтегрування і навіть обчислення значення для трансцендентної функції набагато спрощуються, якщо її наблизити многочленом.

В обох випадках існує два підходи для вибору конкретного методу.

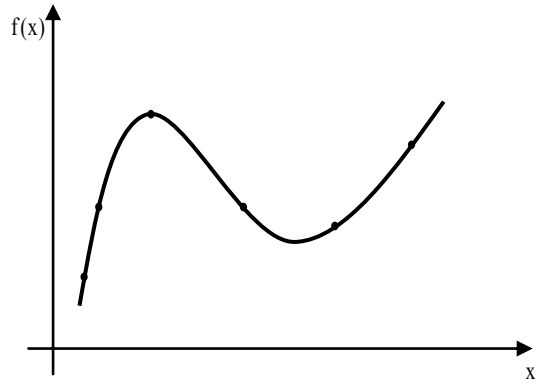


Рис 2.1- Інтерполяція

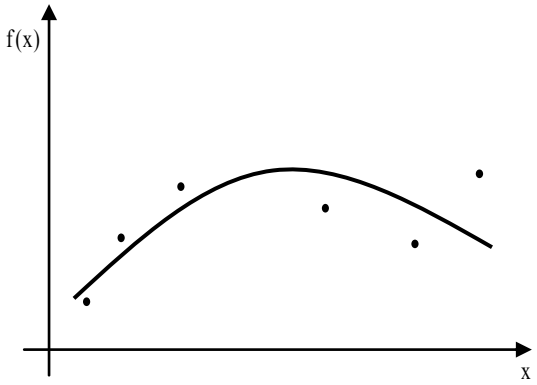


Рис 2.2- Наближення у середньому

Можна поставити вимогу, щоб наближена функції у вузлах співпадала з табличними значеннями (Рис 2.2), і тоді маємо задачу інтерполяції. Можна ж побудувати наближення так щоб воно проходило близько від всіх табличних значень, і не обов’язково через якесь з них (Рис 2.2).

Вибір першого чи другого підходу визначається характером даних, тобто яким чином одержані ці таблиці. Можливі дві ситуації:

а) дані значення (x_i, y_i) містять похибку, яка може бути як випадковою так і іншої природи;

б) дані значення надійні.

Для випадку (б) будується інтерполяційна функція, що проходить через всі дані точки (x_i, y_i) У випадку (а) це може привести до неприємностей:

інтерполяційна функція може в значній мірі посилити випадкові флуктуації даних, в той час як їх необхідно звести до мінімуму. Тому, якщо дані ненадійні то необхідно будувати апроксимуючу функцію.

Розглянемо спочатку задачу інтерполяції, так як історично вона була розв'язана першою. З практичної точки зору найбільш важливим класом для інтерполяції функцій є множина алгебраїчних поліномів. Поліноми мають очевидні переваги - їх легко обчислювати, додавати, множити, а також диференціювати і інтегрувати. Ще одна важлива властивість: якщо c - константа, $p(x)$ - поліном, то поліномами будуть і $p(cx)$, і $p(x+c)$.

Зрозуміло, клас функцій може мати всі вказані властивості, проте не мати задовільних апроксимуючих властивостей. Підставою вважати, що будь яка неперервна функція на замкнутому інтервалі може бути добре наближена деяким поліномом служить результат теорії наближень – апроксимаційна теорема Вейерштрасса. Ця теорема формулюється наступним чином:

Якщо $f(x)$ - неперервна на скінченому замкнутому інтервалі $[a,b]$ функція, то для будь якого $\varepsilon > 0$ існує поліном $p_n(x)$ ступеня $n = n(\varepsilon)$ такий, що

$$\max_{x\in[a;b]} |f(x) - p_n(x)| < \varepsilon$$

Як правило наближення шукається у формі

$$f(x) = \sum_{i=1}^{n} \alpha_i \varphi(x) \qquad (2.1.1)$$

де функції $\varphi_i(x)$, що мають назву базисних або координатних, вважаються відомими і вибираються з міркувань точності та зручності обчислень. Тобто задача полягає у визначенні набору коефіцієнтів a_i. Якщо у якості базису вибрати ступеневі функції

$$\varphi_i(x) = a_i x^{i-1}, i = 1,2,\ldots,$$

то $f(x)$ буде многочленом, коефіцієнти якого a_i можна визначити із рішення СЛАР

$$\sum_{i=1}^{n} a_i x_j^i = y(x_j)\,,\ \mathrm{j} = 0,1,\ldots,\mathrm{n} \qquad (2.1.2)$$

Система (2.1.2) матиме єдине розв'язок, коли її визначник не дорівнює нулю. Цей визначник називається визначником Вандермонда і обчислюється як

$$\det(A) = \begin{vmatrix} 1 & x_1 & x_1^2 & x_1^3 & \ldots\ldots & x_1^{n-1} \\ 1 & x_2 & x_2^2 & x_2^3 & \ldots\ldots & x_2^{n-1} \\ & & & & & \\ 1 & x_n & x_n^2 & x_n^3 & \ldots\ldots & x_n^{n-1} \end{vmatrix} = \prod_{i,\,j=1} (x_i - x_j)$$

З останньої формули випливає, що визначник цієї системи буде дорівнювати нулю тільки у тому випадку коли таблиця містить кратні вузли, тобто коли $x_k = x_m$ при $k \neq m$. Таким чином якщо кратні вузли у таблиці відсутні, то задача має єдине рішення.

Після того як для інтерполяції було використано многочлен ступеня $\leq n-1$, залишається можливість вибору базису в просторі таких багаточленів. Вибір базису з одночленів $1, x, x^2, \ldots x^{n-1}$ приводить до системи лінійних рівнянь, визначники матриць яких хоч і не дорівнюють нулю, але в деяких випадках ці рівняння надзвичайно погано обумовлені. Допустимо, наприклад, що абсциси $\{x_i\}$ розподілені приблизно рівномірно на інтервалі $[0,1]$. Виявляється, що послідовні ступені $1, x, x^2, \ldots x^{n-1}$ майже лінійно залежні на інтервалі $[0,1]$ зокрема тому, що всі вони додатні і їх графіки ідуть від точки $(0,0)$ до $(1,1)$. Саме ця близькість до лінійної залежності робить розв'язок системи при нормальній робочій точності досить складною справою. Крім цього визначення коефіцієнтів з розв'язку СЛАР не є доцільним, так як зі збільшенням кількості вузлів зростає обсяг обчислень і величина похибки.

2.2 Інтерполяційний поліном Лагранжа

У формі запропонованій Лагранжем многочлен (2.1.1) має вигляд

$$f(x) = \sum_{i=1}^{n} y(x_i)\Phi_i(x)\,,$$

де $\Phi_i(x)$ - поліном ступеня не нижче $n-1$, такий що

$$\Phi_i(x) = \delta_i^j = \begin{cases} 1, & \text{i} = \text{j} \\ 0, & \text{i} \neq \text{j} \end{cases} \qquad (2.2.1)$$

Тоді для i-го вузла маємо $f(x_i) = \sum_{k=1}^{n} y(x_i)\delta_i^k = y(x_i)$

Вимозі $\Phi_i(x_j) = 0, \text{i} \neq \text{j}$ відповідає многочлен

$$\Phi_i(x) = Const \times (x - x_0) \times (x - x_1) \times \ldots \times (x - x_n) = Const \times \prod_{i \neq j} (x - x_j)$$

Константу можна визначати з умови $\Phi_i(x_i) = 1$, тобто

$$Const = \frac{1}{\prod_{i \neq j} (x - x_j)},$$

і будуть виконані обидві умови в (1.3).

Таким чином остаточно одержуємо *інтерполяційний поліном Лагранжа,* який дозволяє побудувати многочлен $n-1$ -го ступеня по n вузлах

$$L_n(x) = \sum_{i=1}^{n} y(x_i)\omega_i(x) = \sum_{i=1}^{n} y(x_i) \prod_{i \neq j} \frac{(x - x_j)}{(x_i - x_j)} =$$

$$= y(x_0)\frac{(x - x_2)(x - x_3)\ldots(x - x_n)}{(x_1 - x_2)(x_1 - x_3)\ldots(x_1 - x_n)} +$$

$$+ y(x_1)\frac{(x - x_1)(x - x_3)\ldots(x - x_n)}{(x_2 - x_1)(x_2 - x_3)\ldots(x_2 - x_n)} + \ldots + \qquad (2.2)$$

$$+ y(x_n)\frac{(x - x_1)(x - x_2)\ldots(x - x_{n-1})}{(x_n - x_1)(x_n - x_2)\ldots(x_n - x_{n-1})} +$$

Якщо б можна було провести ідеальні обчислення(без похибки), то значення наближеної функцій у проміжній точці, визначене за допомогою поліному Лагранжа, і з розв'язку СЛАР (2.1.2), співпадали би. Тобто існує тільки один інтерполяційний многочлен, але форм його зображення може бути декілька. Необхідність у різних формах зображення інтерполяційного многочлена обумовлена різними їх властивостями: обсягом обчислень, точністю, можливістю аналізу.

Проведемо оцінку точності інтерполяційного полінома у формі Лагранжа.

Якщо $L_n(x)$ поліном , побудований по значенням функції $y(x)$ у точках $x_i \ (i=1,2,..n)$, то величина похибки визначається різницею між точним та наближеними значеннями $y(x)-L_n(x)$. Введемо допоміжну функцію

$$\varphi(t)=y(t)-L_n(t)-k\omega_n(t), \qquad (2.2.3)$$

де $\omega_n(t)=(t-x_1)(t-x_2)..(t-x_n)$, а $k=const$.

Щоб знайти цю константу, необхідно покласти, що у точці де обчислюється похибка, тобто при $t=x$, $\varphi(t)=0$.

По визначенню функція $\varphi(x)=0$ набуває нульових значень у $n+1$ точках. По теоремі Ролля, якщо функція має на деякому інтервалі два рівні значення, то в цьому інтервалі існує точка , у якій її похідна рівна нулю. Із будови функції (2.2.3) витікає , що похідна $\varphi'(x)$ має n нулів, похідна $\varphi''(x)$ - $n-1$ нулів , і т.д. Після n- кратного диференціювання (2.2.2) з врахуванням, що $L^{(n)}(x)=0$, маємо

$$\varphi^{(n)}(\varsigma)=y^{(n)}(\varsigma)-kn!=0.$$

Тут ς - деяка точка з інтервалу інтерполяції . Таким чином $k=\dfrac{y^{(n)}(\varsigma)}{n!}$ і формула для оцінки точності набуває вигляду:

$$y(x)-L_n(x)=\frac{y^{(n)}(\varsigma)}{n!}\omega_n(x) \qquad (2.2.4)$$

Хоча цей результат важко застосувати на практиці за того, що не відомі ні значення ς, ні величина похідної $y^{(n)}(\varsigma)$, але далі буде показано, як за допомогою (2.2.4) похибка інтерполяції може бути зведена до мінімуму.

2.3 Інтерполяційний поліном Ньютона

Щоб одержати наступну форму інтерполяційного многочлена розглянемо поняття поділеної різниці. Поділена різниця є характеристикою функції, що за

змістом є узагальненою похідною. Якщо похідна у звичайному значенні є локальною характеристикою функції у точці, то узагальнена похідна описує поведінку функції на деякому відрізку. По визначенню значення поділеної різниці нульового порядку є значення самої функції. Поділені різниці першого та другого порядку визначаються як

$$f(x_i; x_j) = \frac{f(x_j) - f(x_i)}{x_j - x_i},$$

$$f(x_i; x_j; x_k) = \frac{f(x_j; x_k) - f(x_i; x_j)}{x_k - x_i}$$

Для поділених різниць вищих порядків використовується рекурентна формула

$$f(x_1, x_2; \ldots; x_k; x_{k+1}) == \frac{f(x_2; \ldots; x_{k+1}) - f(x_1; \ldots; x_k)}{x_k - x_1}$$

Для обчислення поділених різниць дуже зручною є наступна таблиця

x_0 f_0

$$\frac{f_1 - f_0}{x_1 - x_0}$$

$$\frac{1}{x_2 - x_0} \cdot \left[\frac{f_2 - f_1}{x_2 - x_1} - \frac{f_1 - f_0}{x_1 - x_0} \right]$$

x_1 f_1

$$\frac{f_2 - f_1}{x_2 - x_1}$$

x_2 f_2

$$\frac{1}{x_3 - x_1} \cdot \left[\frac{f_3 - f_2}{x_3 - x_2} - \frac{f_2 - f_1}{x_2 - x_1} \right]$$

$$\frac{f_3 - f_2}{x_3 - x_2}$$

x_3 f_3

Аналогічні формули можна одержати для різниць побудованих по точках: $x_n, x_{n-1}, x_{n-2,}\ x_{n-3}, \ldots$

Обчислення за допомогою таблиць звичайно використовують для різниць відносно невисоких порядків, а для машинних розрахунків більш зручною є наступна формула

$$f(x_1; x_2; \ldots; x_n) = \sum_{j=1}^{n} \frac{f(x_{j)}}{\prod_{i \neq j}^{n} (x_j - x_i)} \qquad (2.3.1)$$

Інтерполяційний поліном у формі Ньютона досить просто отримати з визначення поділених різниць. З формули для поділеної різниці першого порядку

$$f(x; x_0) = \frac{f(x) - f(x_0)}{x - x_0}$$

знаходимо

$$f(x) = f(x_0) + (x - x_0) f(x; x_0) (\ ).$$

Невідомий на цьому кроці множник у правій частині ( ) можна знайти, використавши поділену різницю другого порядку

$$f(x; x_0; x_1) = \frac{f(x; x_0) - f(x_0; x_1)}{x - x_1}.$$

Звідки $f(x; x_0) = f(x_0; x_1) + (x - x_1) f(x; x_0; x_1)$ і формула ( ) набере вигляду

$$f(x) = f(x_0) + (x - x_0) f(x; x_0) + (x - x_0)(x - x_1) f(x; x_0; x_1)$$

Повторивши цю операцію n разів маємо остаточний результат

$$f(x) = f(x_0) + (x - x_0) f(x; x_0) + (x - x_0)(x - x_1) f(x; x_0; x_1) + \ldots$$
$$+ (x - x_0)(x - x_1) \ldots (x - x_{n-1}) f(x; x_0; x_1, \ldots; x_n)$$

Якщо значення функції визначені в рівновіддалених вузлах, то в цьому випадку поділені різниці називаються скінченими різницями.

Ітераційна схема Ейткіна використовується тоді коли необхідно знайти значення функції тільки в одній точці з заданою точністю. На першому кроці наближення знаходиться шляхом лінійної інтерполяції по двох точках x_0, x_m

$$L_{m1}(x) = \frac{1}{x_m - x_0}\left[L_{00}(x)(x_m - x) - L_{m0}(x)(x_0 - x)\right]$$
$$L_{i0}(x) = f(x_i)$$

Наступне квадратичне наближення будується по трьох точках $x_0,\ x_1, x_m$ (рис 2.3)

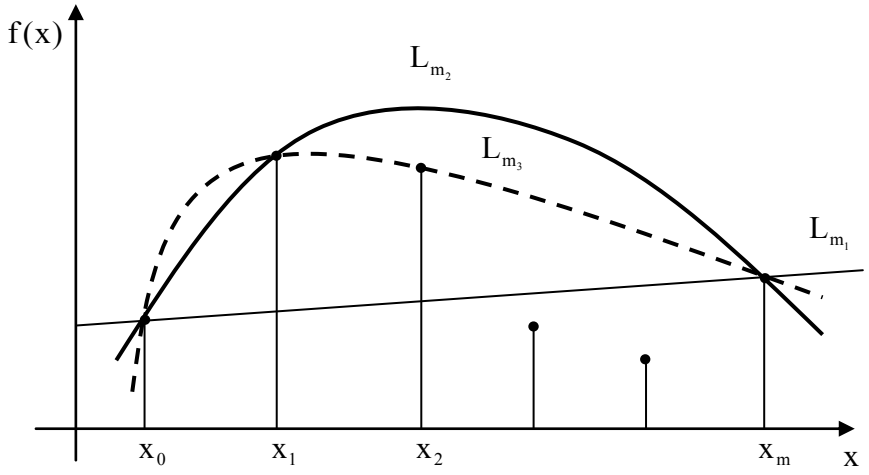


Рис 2.3.- Підвищення ступеня інтерполяції в схемі Ейткена

$$L_{m2}(x) = \frac{1}{x_m - x_1}\left[L_{11}(x)(x_m - x) - L_{m1}(x)(x_1 - x)\right]$$

Для полінома k-го ступеня маємо

$$L_{mk}(x) = \frac{1}{x_m - x_{k-1}}\left[L_{k-1k-1}(x)(x_m - x) - L_{mk-1}(x)(x_{k-1} - x)\right] \qquad (2.3.2)$$

Перехід від k-го наближення до $(k+1)$-го потребує попереднього обчислення тільки величин

$$L_{kk}(x), L_{kk-1}(x), \ldots, L_{k2}(x), L_{k1}(x),$$

оскільки всі інші значення поліномів у(2.3.2) були визначені на попередніх ітераціях. Після кожного чергового наближення перевіряється умова зупинки по досягненню потрібної точності.

$$\left|L_{mn}(x) - L_{mn-1}(x)\right| \le \varepsilon$$

На практиці цей метод використовується як правило для поліному не вище четвертого ступеня.

Зі збільшенням кількості вузлів інтерполяції зростає ступінь інтерполяційного многочлена і зростає похибка округлення. Тому точність наближення у цьому випадку зменшується. Обчислення наближення за допомогою поліномів Лагранжа дають найменшу похибку і потребують мінімальної кількості операцій. Також його застосування для наближення декількох функцій в одній і тій же точці є оптимальним. Дійсно, досить спочатку обчислити многочлен $\omega_i(x)$ в (2.2.2), а далі результат отримуємо простим під сумуванням. Недоліком цієї інтерполяційної формули є те, що додання хоча б одного вузла в таблицю потребує перебудови всього многочлена. В інтерполяційній формулі Ньютона у цій ситуації досить перерахувати тільки останній додаток. Звичайно формула Лагранжа застосовується для практичних розрахунків, а Ньютона - як основа для рішення інших задач: диференціювання, інтегрування, тощо.

2.4 Поліноми Чебишева

Система функцій $\varphi_i(x)$, визначених на інтервалі $[a;b]$ називається ортогональною з ваговою функцією $\gamma(x)$, якщо виконується умова

$$\int_a^b \gamma(x)\varphi_m(x)\varphi_n(x)dx = \begin{cases} \neq 0, m = n \\ = 0, m \neq n \end{cases}$$

Якщо при цьому

$$\int_a^b \gamma(x)\varphi_m(x)\varphi_n(x)dx = \delta_{mn}$$

,

то така система називається ортонормованою.

Ортогональні функції мають важливе значення в чисельних методах, зокрема в задачах наближення функцій.

Умові ортогональності задовольняють поліноми Чебишева $T_n(x)$, які можуть бути визначенні за допомогою рекурентної формули

$$T_{n+1}(x) = 2xT_n(x) - T_{n-1}(x), \qquad (2.4.1)$$

з граничними умовами $T_0(x) = 1,\ T_1(x) = x$.

За допомогою цієї формули легко одержати поліноми наступних ступенів

$T_2(x) = 2x^2 - 1,$

$T_3(x) = 4x^3 - 3x,$

$T_4(x) = 8x^4 - 8x^2 + 1,$

Крім формули (2.4.1) поліноми Чебишева можуть бути визначені іншим чином.

Якщо у відомій тригонометричній формулі.

$$2\cos\alpha\cos\beta = \cos(\alpha+\beta) - \cos(\alpha-\beta)$$

або $$\cos(\alpha+\beta) = 2\cos\alpha\cos\beta - \cos(\alpha-\beta),$$

позначити $\alpha = n\theta$, а $\beta = \theta$, то

$$\cos[(n+1)\theta] = 2\cos\theta\cos(n\theta) - \cos[(n-1)\theta]. \qquad (2.4.2)$$

Позначення ж $\theta = \arccos x, x = \cos x$ приводять (2.4.2) до рекурентної формули для поліномів Чебишева у тригонометричній формі

$$\cos[(n+1)\arccos x] = 2x\cos[(n+1)x] - \cos[(n-1)\arccos x].$$

З тригонометричного зображення

$$T_n(x) = \cos(n\arccos x) \quad (2.4.3)$$

безпосередньо визначаються граничні умови для рекурентної формули

$T_0(x) = \cos(0) = 1,\ T_1(x) = \cos(\arccos x) = x$

Тригонометричне зображення є зручним для визначення нулів і екстремумів поліномів. З (2.4.3) легко одержати формула для нулів поліномів Чебишева

$$T_n(x) = \cos(n\arccos x) = 0,\ n\arccos x = \frac{\pi}{2} + \pi k, k = 0,1,\ldots,n$$

$$x_k = \cos\left[\frac{\pi}{2n}(2k+1)\right]$$

Аналогічно можемо одержати формулу для точок екстремумів. Так

$$[\cos(n\arccos x)]' = -\frac{n}{\sqrt{1-x^2}}\sin(n\arccos x) = 0,$$

$$n\arccos x = \pi m, \quad m = 0,1,2,\ldots,n$$

і тоді $x_k = \cos\left(\frac{\pi m}{n}\right),\ m = 0,1,2,\ldots,n$

Поліноми Чебишева є ортогональними на інтервалі $[-1;1]$ з ваговою функцією $\gamma(x) = \frac{1}{\sqrt{1-x^2}}$.

Якщо в умові ортогональності $\int_{-1}^{1} \frac{T_m(x)T_m(x)}{\sqrt{1=x^2}}dx$ зробити заміну

$$\theta = \arccos x, \quad d\theta = \frac{n}{\sqrt{1-x^2}}dx$$

і врахувати, що $T_n = \cos n\theta$, то

$$\int_{-1}^{1} \frac{T_m(x)T_m(x)}{\sqrt{1=x^2}}dx = n\int_{\pi}^{0} \cos n\theta \cos m\theta d\theta = \begin{cases} \pi, m = n = 0 \\ \frac{\pi}{2}, m = n \neq 0 \\ 0, m \neq n \end{cases}.$$

Отриманий результат показує, що умова ортогональності виконується. Якщо поліном Чебишева визначається на дискретній множині точок з інтервалу $[-1;1]$, то умова ортогональності набуває вигляду

$$\sum_{i=1}^{N} T_m(x_i)T_m(x_i) = \begin{cases} N, m = n = 0 \\ \frac{N}{2}, m = n \neq 0 \\ 0, m \neq n \end{cases}$$

Використання поліномів Чебишева в обчислювальній практиці можна показати на практиці наступних задач.

Наближення функцій. Якщо наближення функції f шукається у вигляді

$$f(x) = \sum_{k=0}^{n} \alpha_k \varphi_k(x),$$

то за допомогою метода найменших квадратів, у якому коефіцієнти α_k знаходяться з умови

$$\sum_{i=1}^{N} \left[\sum_{k=0}^{n} \alpha_k \varphi_k(x_i) - f(x) \right]^2 \to \min$$

задача зводиться до системи лінійних алгебраїчних рівнянь

$$\sum_{k=0}^{n} A_{mk}\alpha_k = B_m,$$

де $A_{mk} = \sum_{i=1}^{N} \varphi_m(x_i)\varphi_{mk}(x_i)$, $B_m = \sum_{i=1}^{N} \varphi_m(x_i) f(x_i)$.

В силу властивості ортогональності функцій $\varphi_m(x)$ в матриці А відмінними від нуля будуть тільки діагональні елементи A_{mk} і система має безпосереднє рішення:

$$\alpha_m = \frac{B_m}{A_{mm}} .$$

Мінімізація похибки інтерполяції. Має місце теорема, яка приводиться тут без доведення.

Із всіх поліномів ступеня n з одиничним коефіцієнтом при старшому ступені найменше відхилення від нулю мають поліноми Чебишева.

Очевидно, що поліноми Чебишева зображені у вигляді $\frac{T_n(x)}{2^{n-1}}$ мають одиничний коефіцієнт при старших ступенях і справедлива оцінка

$$\max|P_n(x)| \geq \max\left|\frac{T_n(x)}{2^{n-1}}\right|,$$

де $P_n(x)$ – довільний поліном з одиничним коефіцієнтом при x^n. Очевидно, що многочлен буде визначеним у формі

$$P_n(x) = const \times (x - x_1)(x - x_2)...(x - x_n) = cont \times \omega_n(x)$$

коли будуть відомі всі його нулі x_i $(i = 1,2,...n)$. Тоді з оцінки похибки інтерполяційного полінома у формі Лагранжа

$$f(x) - L_n(x) = \frac{f^{(n)}(\varsigma)}{n!}\omega_n(x),$$

витікає наступний результат: похибка інтерполяції буде мінімальною, якщо у якості вузлів інтерполяції взяти нулі поліномів Чебишева.

Економізація обчислення ступеневих рядів. Нехай необхідно обчислити значення наступної суми

$$S = a_0 + a_1 x + a_2 x^2 + a_3 x^3 + ... + a_n x_n$$

Ступені x моменту за допомогою зображення поліномів Чебишева в алгебраїчній формі виразити у наступному вигляді

$$x^0 = T_0, \qquad x^1 = T_1, \qquad x^2 = \frac{1}{2}(T_2 + T_0)$$

$$x^3 = \frac{1}{4}(T_3 + T_1), \qquad x^4 = \frac{1}{8}(T_4 + 4T_3 + T_1)$$

і тоді задача обчислення $S = S(x)$ зводиться до обчислення $S = S(T_n)$. При однакових вимогах на точність обчислювань такий підхід дозволяє у сумі ( ) взяти меншу кількість додатків. Наприклад, при обчисленні:

$$S(x) = \ln(1+x) = x - \frac{x^2}{2} + \frac{x^3}{3} - \frac{x^4}{4} + \ldots$$

при збереженні перших трьох членів при $x = 1$ маємо похибку $\Delta = 0{,}25$ (для знакопочередних рядів похибка не перевищує величини по модулю першого відкинутого члена). Якщо тепер обчислити

$$S(T) \approx -\frac{1}{4}T_0 + \frac{5}{4}T_1 - \frac{1}{4}T_2$$

і врахувати, що $T_n(1) = 1$, то у цьому випадку необхідну точність одержується уже при збереженні тільки двох перших членів. Тобто такий підхід дає можливість скоротити об’єм обчислювань.

2.5 Метод найменших квадратів

На відміну від інтерполяційних методів даний підхід дає таке наближення у якому отримане значення функції не обов’язково проходить через задані точки. Наближена функція знаходиться з умови щоб сума квадратів відхилень наближеної функції $f(x)$ від заданої $y(x_i)$ була найменша, тобто задача зводиться до визначення мінімуму функції

$$\Phi = \sum_{j=1}^{n} \left[f(x_j) - y(x_j) \right]^2$$

Коли наближення шукається у формі (2.1.1), то задача полягає у визначенні коефіцієнтів α, що мінімізують функцію відхилення

$$\Phi = \sum_{j=1}^{n}\left[\sum_{i=1}^{M}\alpha_i\varphi_i(x_j) - y(x_j)\right]^2$$

Необхідні умови мінімуму функції $\frac{\partial\Phi}{\partial\alpha_k} = 0,\ \mathrm{k} = 1,2,..,\mathrm{M}$ приводять до системи лінійних рівнянь:

$$\sum a_{ki}\alpha_i = b_k\ ,$$

де елементи матриці та компоненти вектора правої частини визначаються як

$$a_{ki} = \sum_{j=1}^{n}\varphi_i(x_j)\varphi_k(x_j)\ \ b_k = \sum_{j=1}^{n} y(x_j)\varphi_k(x_j)$$

У випадку наближення лінійною функцією $f(x) = \alpha_1 + \alpha_2 x$ параметри наближення визначаються шляхом рішення задачі

$$\Phi = \sum_{j=1}^{n}\left[\alpha_1 + \alpha_2 x_j - y(x_j)\right]^2 \to \min,$$

що приводять до системи лінійних рівнянь:

$$\begin{cases} \alpha_1 n + \alpha_2\sum_{j=1}^{n} x_j = \sum_{j=1}^{n} y(x_j) \\ \alpha_1\sum_{j=1}^{n} x_j + \alpha_2\sum_{j=1}^{n} x_j^2 = \sum_{j=1}^{n} x_j y(x_j) \end{cases}$$

відносно невідомих α_1, α_2.

Кількісною мірою наближення виду вибраної функції до дійсної функціональної залежності є теоретичне кореляційне відношення

$$\eta^2 = \frac{\sum_{k=1}^{n}\left[f(x_k) - m_f\right]^2}{\sum_{k=1}^{n}(y_k - m_f)^2}, \qquad m_f = \frac{1}{n}\sum_{k=1}^{n} y_k$$

Ця величина змінюється у межах $0 \le \eta^2 \le 1$. Для граничних випадків маємо

$\eta^2 \approx 0$ - функціональна залежність між x та y відсутня;

$\eta^2 \approx 1$ - функціональна залежність вибрана ідеально. На практиці вважається, що значення $\eta^2 \geq 0{,}5$ є цілком достатнім для прийняття вибраного наближення.

Якщо зі збільшенням заданих точок точність наближення шляхом інтерполяції зменшується то при використанні МНК навпаки зростає.

Контрольні питання.

Яким чином з декількох наближень вибрати найкраще?

Що відбувається з наближенням по МНК зі збільшенням вузлів у таблиці?

Чи може таблиця у випадку наближення по МНК мати кратні вузли?

У яких випадках застосовується наближення по МНК, а у яких будується інтерполяційний поліном?

Чим визначається розмір СЛАР при наближенні по МНК?

У якому з методів інтерполяції чи МНК можна підвищити точність наближення і яким чином?

На рис. 1.4 представлені результати наближення функції $f(x) = xe^x$ на інтервалі [1; 3] по десяти рівновіддалених вузлах. Лінійна функція має вигляд $f_1(x) = -32{,}424 + 26{,}894x$ і їй відповідає значення $\eta^2 = 0{,}8864$. Для квадратичної залежності маємо $f_2(x) = 27{,}682 - 40{,}028x + 16{,}731x^2$ і $\eta^2 = 0.9948$.

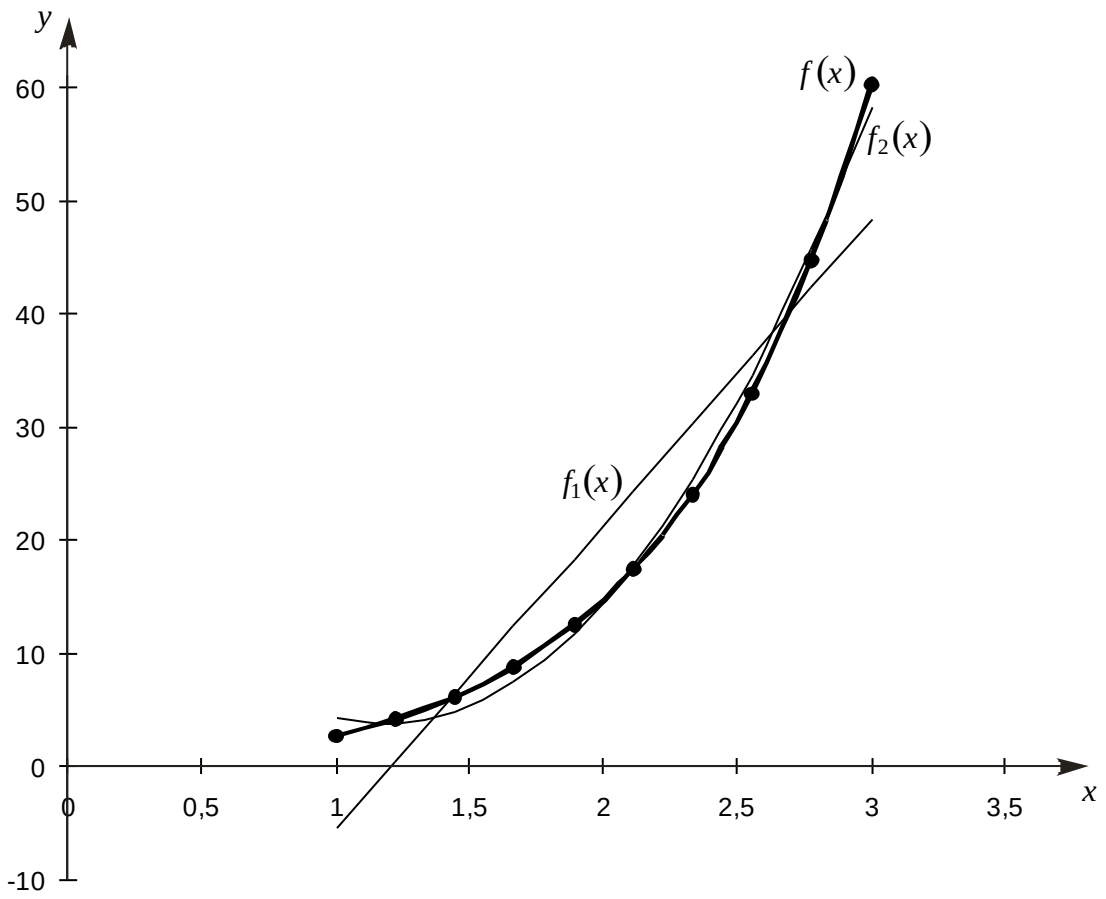


Рис. 1.4.- Наближення по методу найменших квадратів

2.6 Наближення сплайн - функціями

Якщо функція $f(x)$ дуже складно змінюється на різних ділянках відрізку $[a;b]$, або цей відрізок досить великий , то важко апроксимувати її з хорошою точністю на всьому $[a;b]$ за допомогою одного інтерполяційного поліному $L_n(x)$. Також зростання кількості вузлів інтерполяції приводить до зростання ступеня $L_n(x)$, а значить, і до зростання похибки. Що уникнути цих труднощів, можна розбити відрізок ні якісь частини. На кожній частині побудувати свій інтерполяційні поліноми, а потім «склеїти» їх в точках розбиття. Приблизно така ідея закладена в сплайн – інтерполяції . Креслярі здавна використовували механічні сплайни, що представляють собою гнучкі рейки з якого не будь пружного матеріалу. Механічні сплайни закріпляють і підвішують ваги в точках інтерполяції, що отримали історичну назву вузли.

Сплайн приймає форму, яка мінімізує його потенціальну енергію, і в теорії балок установлено, що ця енергія пропорційна інтегралові по довжині дуги від квадрату кривизни сплайна.

Якщо сплайн позначити функцією $s(x)$, то при малих нахилах друга похідна $s''(x)$ наближено дорівнює кривизн, а диференціал дуги - dx. Таким чином, енергія подібного лінеаризованого сплайну пропорційна інтегралу $\int (s''(x))^2 dx$.

Оскільки механічний сплайн не, руйнується, то слід чекати, що і s, і s' неперервні на $[x_1; x_n]$. З елементарної теорії балок випливає, що $s(x)$ є кубічним поліномом між кожною сусідньою парою вузлів і що сусідні поліноми з'єднуються неперервно, так як ф їх перші і другі похідні.

Кубічна сплайн- функція, що задовольняє умовам $s''(x_1) = s''(x_n) = 0$, називається природнім кубічним сплайном. Доведено, що вона є єдиною функцією, яка має мінімальну кривизну, серед всіх функцій , що інтерполюють дані точки мають квадратично інтегруєму другу похідну. В цьому сенсі кубічний сплайн є сама гладка з функцій, що інтерполюють дані точки.

У загальному випадку для завдання поліному третього ступеня необхідно мати чотири коефіцієнти. Якщо таблиця складається з n вузлів, то інтервалів між ними буде $n-1$ і треба визначити $4(n-1)$ невідому. Той факт що функція та її перша та друга похідні неперервні в $n-2$ вузлах рівносильний $3(n-2)$ умовам. Крім того вимога $s(x_i) = f(x_i)$ дає ще n умов. Потрібні ще дві додаткові умови, щоб однозначно визначити сплайн. Якщо ці умови мають вигляд $s''(x_1) = s''(x_n) = 0$, то такий сплайн називається природнім.

Побудова кубічного сплайну- простий і чисельно стійкий процес.. Розглянемо підінтервал $[x_i; x_{i+1}]$ і нехай

$$h_i = x_{i+1} - x_i; \qquad t = \frac{x - x_i}{h_i}; \quad \bar{t} = 1 - t.$$

Коли x пробігає цей підінтервал, t змінюється від 0 до 1, а $\bar{t}$ від 1 до 0. Представимо сплайн на цьому під інтервалі у формі

$$s(x) = y_{i+1} \cdot t + y_i \cdot \bar{t} + h_i^2 \cdot \left[\sigma_{i+1} \cdot \left(t^3 - t\right) + \sigma_i \cdot \left(\bar{t}^3 - \bar{t}\right)\right],$$

де σ_i і σ_{i+1}- деякі константи, які належить визначити. Перші два члена цього виразу відповідають стандартній лінійній інтерполяції, а вираз у квадратних дужках є кубічною добавкою, яка забезпечує додаткову гладкість. На кінцях підінтервалу, ця добавка дорівнює нулю, так що

$s(x_i) = y_i$, $s(x_{i+1}) = y_{i+1}$

Таким чином $s(x)$ інтерполює задані значення, незалежно від чисел σ_i.

Таким чином задача зводиться до розв'язку СЛАР з матрицею загального виду.

З врахуванням того, $t' = \frac{1}{h_i}$, а $\bar{t}' = -\frac{1}{h_i}$ для перших двох похідних маємо

$$s'(x) = \frac{y_{i+1} - y_i}{h_i} + h_i\left[\left(3t^2 - 1\right)\sigma_{i+1} \cdot -\left(3\bar{t}^2 - \bar{t}\right)\sigma_i\right] \qquad (\ )$$

$$s''(x) = 6t\sigma_{i+1} + 6\bar{t}\sigma_i$$

З останньої формули негайно випливає, що $s''(x)$ є лінійною функцією, яка інтерполює значення $6\sigma_{i+1}$ і $6\sigma_i$.

Таким чином представлення сплайн - функції у формі ( ) дозволяє виконати умови неперервності самої функції та її другої похідної у вузлах x_i автоматично. Умова неперервності першої похідної у вузлі x_i , обчислених по формулі ( ) на підінтервалах $[x_{i-1}; x_i]$ при $t = 1$ і $[x_i; x_{i+1}]$ при $t = 0$ приводить до наступної системи рівнянь для визначення коефіцієнтів σ_i

$$h_{i-1} \cdot \sigma_{i-1} + 2 \cdot \left(h_{i-1} + h_i\right) \cdot \sigma_i + h_i \cdot \sigma_{i+1} = \Delta_i - \Delta_{i-1}, \qquad (2.6.1)$$

де $i = 2,3,\ldots,n-1$, $\Delta_i = \frac{y_{i+1} - y_i}{x_{i+1} - x_i}$

Матриця цієї системи має трьох діагональну структуру і як буде показано у розділі 5.1 кількість операції для її розв'язку набагато менша, ніж у випадку матриці загального вигляду.

При зміні значення функції хоча б у одному вузлу, а також при вилученні чи додаванні вузлів систему (2.6.1) треба розв'язувати знову. На практиці існує ряд задач, у яких наближена функція повинна відповідати вимогам гладкості (неперервності похідних), але інтерполяційна таблиця міняється настільки швидко, що кожен раз розв'язувати СЛАР навіть спрощеної структури (2.6.1) технічно неможливо. У цьому випадку використовують так званий сплайн, що згладжує, запис якого у параметричній формі має вигляд

$$x(t) = ((a_3 \cdot t + a_2) \cdot t + a_1) + a_0 \ ,$$
$$y(t) = ((b_3 \cdot t + b_2) \cdot t + b_1) + b_0 \quad t \in (0;1)$$

де коефіцієнти обчислюються за формулами

$$a_3 = \frac{1}{6}(-x_{i-1} + 3x_i - 3x_{i+1} + x_{i+2}),$$

$$a_2 = \frac{1}{2}(x_{i-1} - 2x_i + x_{i+1})$$

$$a_1 = \frac{1}{2}(-x_{i-1} + x_{i+1}), \qquad (2.6.2)$$

$$a_0 = \frac{1}{6}(x_{i-1} + 4x_i + x_{i+1})$$

Формули для визначення коефіцієнтів b_i мають вигляд аналогічний (2.6.2)

Така форма запису гарантує мінімальну кількість операцій для обчислення многочлена , визначення коефіцієнтів не потребує розв'язку СЛАР. При зміні однієї точки (x_i, y_i) у таблиці значення коефіцієнтів зміниться тільки на чотирьох відрізках. Безпосередніми обчисленнями можна переконатися, що вимоги неперервності функцій $x(t), y(t)$ та їх перших двох похідних будуть

виконані, але значення наближеної функції у вузлах не будуть співпадає з табличним.

2.6 ЛАБОРАТОРНА РОБОТА № 1 **Наближення за допомогою інтерполяційного поліному Лагранжа**

Завдання до лабораторної роботи. Для функції, яка задається у вигляді таблиці, знайти її наближене значення у довільній точці інтервалу.

Методичні вказівки до виконання лабораторної роботи.

На практиці знаходження наближеного значення функції за допомогою інтерполяційного поліному Лагранжа зводиться до програмної реалізації формули

$$f(x) = \sum_{i=1}^{n} y(x_i) \prod_{i \neq j} \frac{(x - x_j)}{(x_i - x_j)}, \qquad (1.11)$$

вихідними даними в якій є інтерполяційна таблиця, тобто два одновимірних вектори x, y. Звичайно ця таблиця читається з файлу даних, але у цій роботі для того, щоб мати можливість оцінити точність результату, таблиця формується як значення заданої функції в довільних точках вказаного інтервалу.

Введемо наступні позначення (ідентифікатори) для величин, які будуть необхідні для виконання програми:

s – наближене значення функції;

z – точка, в якій обчислюється наближення;

p – допоміжна змінна для обчислення добутку в формулі (1.11).

Обчислення проводяться у формі подвійного циклу з параметрами I (зовнішній цикл) та j (внутрішній цикл), які змінюються в межах від 1 до n. При цьому внутрішній цикл повинен виконуватись тільки у тому випадку колі $j \neq i$, що забезпечується застосуванням відповідного умовного оператора. Змінній s

попередньо треба надати значення нуль, а змінній p треба надавати значення одиниці перед початком внутрішнього циклу.

Контрольні питання.

Що таке інтерполяція?

Які існують обмеження на таблицю інтерполяції?

Поліном якого ступеня можна побудувати по n вузлам?

Що відбувається з точністю наближення зі збільшенням вузлів інтерполяції?

Як перевірити правильність обрахунків у випадку задачі інтерполяції?

В яких випадках використовується інтерполяційний поліном в формі Лагранжа, в яких – у формі Ньютона?

Варіанти завдань до лабораторної роботи.

Варіант	Функція	Інтервал
1	$f(x)=x\sin(x)$	$0\le x\le\pi$
2	$f(x)=x^2\cos(x)$	$-\pi\le x\le\pi$
3	$f(x)=xe^{-x}+x^2$	$-1\le x\le 1$
4	$f(x)=x^2\ln x-x$	$1\le x\le 2$
5	$f(x)=\sin(x)/x-x$	$1\le x\le 2$
6	$f(x)=x^3\sin(x+1)$	$-2\le x\le 1$
7	$f(x)=2x^2+\ln(x+1)$	$0\le x\le 2$
8	$f(x)=1/x^2-e^{-x}$	$1\le x\le 3$
9	$f(x)=2x\sin(x)/(x^2+1)$	$-2\le x\le 2$
10	$f(x)=\ln(x+2/(x+1)$	$0\le x\le 1$
11	$f(x)=(\cos x+2)(x+1)$	$0\le x\le 2$
12	$f(x)=\sin(x^2+1)$	$-2\le x\le 2$
13	$f(x)=\sqrt{x}\cos x$	$1\le x\le 3$
14	$f(x)=\sqrt{x}/(1+\sqrt{x})$	$1\le x\le 3$
15	$f(x)=x\ln x+x^2$	$2\le x\le 4$

16	$f(x)=x\sin(x)+\cos x^2$	$1\le x\le 3$
17	$f(x)=2\sin x/(x^2+1)$	$-2\le x\le 2$
18	$f(x)=\ln(x+2)/(x+1)$	$0\le x\le 1$
19	$f(x)=2x^2+\ln(x+1)$	$0\le x\le 2$
20	$f(x)=1/x^2-e^x$	$1\le x\le 3$
21	$f(x)=\sin x/x-x$	$1\le x\le 2$
22	$f(x)=x^3\sin(x-1)$	$-2\le x\le 1$
23	$f(x)=xe^x+x^2$	$-1\le x\le 1$
24	$f(x)=x^2\ln x-x$	$1\le x\le 2$
25	$f(x)=x\ln(x^2+1)$	$-1\le x\le 2$

2.7 ЛАБОРАТОРНА РОБОТА № 2 **Наближення за допомогою ітераційної схеми Ейткіна**

Завдання до лабораторної роботи. Для функції, яка задається у вигляді таблиці (по даним для ЛР № 1) знайти її наближене значення у довільній точці інтервалу.

Методичні вказівки до виконання лабораторної роботи.

Щоб зберегти відповідність у позначеннях у розрахунковій формулі Ейткіна у програмі слід почати нумерацію вузлів не з одиниці, а з нуля. Замість одновимірного масиву y введемо двовимірний масив L, описаний як

```
Var  L: Array [0..100, 0..100,] Of Real;
```

Початкове наближення можна виконати одночасно з читанням даних з файлу.

```
For i:=0 to n do  Readln(mm, x[i],' ',L[i,0]));
```

Програмно ітераційний процес звичайно має форму циклу Repeat – Until. Умовою виходу з нього є виконання вимоги точності, яка у програмному варіанті має вигляд

Until Del<=eps;, де Del:=Abs(L[n,k]-L[n,k-1]); обчислюється раніше, а eps визначається користувачем.

При відлагодженні таких програмних фрагментів, щоб запобігти зациклювання, корисним є наступний прийом. Введемо лічильник ітерацій, і коли його значення стане рівне граничному, забезпечимо закінчення циклу умовним оператором

If k=n Then del:=0;

Граничне значення лічильника, як правило визначається специфікою задачі, хоча взагалі може бути довільною величиною. В нашому випадку це буде вичерпання всіх значень таблиці. В тілі циклу по ітераціях, де ступінь полінома k змінюється від 1 до n(якщо раніше не спрацює вихід по точності) міститься внутрішній цикл по точкам таблиці. В останньому параметр циклу змінюється від k до n і відповідає індексу m у формулі метода

$$L_{mk}(x) = \frac{1}{x_m - x_{k-1}}\left[L_{k-1k-1}(x)(x_m - x) - L_{mk-1}(x)(x_{k-1} - x)\right].$$

Контрольні питання.

Що таке інтерполяція?

Які існують обмеження на таблицю інтерполяції?

Поліном якого ступеня можна побудувати по n вузлам?

Що відбувається з точністю наближення зі збільшенням вузлів інтерполяції?

Як перевірити правильність обрахунків у випадку задачі інтерполяції?

В яких випадках використовується інтерполяційний поліном в формі Лагранжа, в яких – у формі Ньютона?

2.8 ЛАБОРАТОРНА РОБОТА № 3 **Метод найменших квадратів**

Завдання до лабораторної роботи. Знайти наближення функції, яка задається у вигляді таблиці (по даним для ЛР №1) лінійною $f(x) = \alpha_0 + \alpha_1 x$ і квадратичною $f(x) = \alpha_0 + \alpha_1 x + \alpha_1 x^2$ залежностями. За допомогою теоретичного кореляційного відношення оцінити ступінь близькості вибраних залежностей до дійсної функціональної залежності. Дати графічну інтерпретацію результатів.

Методичні вказівки до виконання лабораторної роботи.

В математичному плані дана задача складається з формування системи лінійних рівнянь і подальшому її розв'язку У прикладі, що був розглянутий наведеному в теоретичній частині для випадку лінійного наближення, для елементів матриці і компонентів вектора правої частини наведені явні вирази. Очевидно, що найменша зміна або ускладнення вигляду базисних функцій потребують виводу нових формул. Тому на практиці звичайно прийнято формувати СЛАР за допомогою загальних формул

$$a_{ki} = \sum_{j=1}^{n} \varphi_i(x_j)\varphi_k(x_j),\ b_k = \sum_{j=1}^{n} y(x_j)\varphi_k(x_j) \qquad (1.12)$$

Для використання формул ( ) потрібно лише попередньо провести нумерацію базисних функцій для всіх точок таблиці .Для наближення поліномами маємо

$$\varphi_0(x_j) = 1,\ \varphi_1(x_j) = x_j,\ \varphi_2(x_j) = x_j^2,\ j = 1,2,\ldots,N_e.$$

Якщо позначити nb – довжина базису, а ne- число точок у таблиці , то формування СЛАР матиме вигляд такого фрагменту програми, який доцільно оформити як окрему процедуру

```
For m:=0 to nb do
Begin  bm[m]:=0;
    For j:=1 to ne do bm[m]:=bm[m]+fb[m,j]*y[j];
     For k:=0 to nb do
     Begin
          ae[m,k]:=0;   bm[m]:=0;
```

```
            For j:=1 to ne do ae[m,k]:=ae[m,k]+fb[m,j]*fb[k,j];
        End;
End;
```

Після формування СЛАР, так як її порядок не високий, розв'язок доцільно провести методом Крамера. Для цього обчислюються головний і допоміжні визначники. Для випадку трьох невідомих маємо

$$\Delta = \begin{vmatrix} a_{00} & a_{01} & a_{02} \\ a_{10} & a_{11} & a_{12} \\ a_{20} & a_{22} & a_{22} \end{vmatrix}, \quad \Delta_0 = \begin{vmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \\ a_{31} & a_{32} & a_{33} \end{vmatrix}, \quad \Delta_1 = \begin{vmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \\ a_{31} & a_{32} & a_{33} \end{vmatrix}, \quad \Delta_2 = \begin{vmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \\ a_{31} & a_{32} & a_{33} \end{vmatrix},$$

$$\Delta = a_{00}a_{11}a_{22} + a_{01}a_{12}a_{20} + a_{10}a_{21}a_{02} - a_{20}a_{11}a_{02} - a_{21}a_{12}a_{00} - a_{10}a_{01}a_{22};$$

```
d:=a[0,0]*a[1,1]*a[2,2]+a[0,1]*a[1,2]*a[2,0]+a[1,0]*a[2,1]*a[0,2]-
    a[2,0]*a[1,1]*a[0,2]- a[1,0]*a[0,1]*a[2,2]-a[2,1]*a[1,2]*a[0,0];
```

$$\Delta_0 = b_0a_{11}a_{22} + b_2a_{01}a_{12} + b_1a_{21}a_{02} - b_2a_{11}a_{02} - b_1a_{01}a_{22} - b_0a_{21}a_{12};$$

```
d0:=b[0]*a[1,1]*a[2,2]+a[0,1]*a[1,2]*b[2]+b[1]*a[2,1]*a[0,2]-
    b[2]*a[1,1]*a[0,2]-b[1]*a[0,1]*a[2,2]-b[0]*a[2,1]*a[1,2];
```

$$\Delta_1 = b_1a_{00}a_{22} + b_0a_{12}a_{20} + b_2a_{10}a_{02} - b_1a_{20}a_{02} - b_0a_{10}a_{22} - b_2a_{12}a_{00};$$

```
d1:=a[0,0]*b[1]*a[2,2]+ b[0]*a[1,2]*a[2,0]+a[1,0]*b[2]*a[0,2]-
    a[2,0]*b[1]*a[0,2]- b[0]*a[1,0]*a[2,2]-b[2]*a[1,2]*a[0,0];
```

$$\Delta_2 = b_2a_{00}a_{11} + b_1a_{01}a_{20} + b_0a_{10}a_{21} - b_0a_{20}a_{11} - b_2a_{10}a_{01} - b_1a_{00}a_{21}$$

```
d2:=a[0,0]*a[1,1]*b[2]+ a[0,1]*b[1]*a[2,0]+a[1,0]*a[2,1]*b[0]-
    a[2,0]*a[1,1]*b[0]- a[1,0]*a[0,1]*b[2]-a[0,0]*a[2,1]*b[1].
```

Компоненти невідомих визначаються з формули

$\alpha_i = \Delta_i / \Delta$.

Для випадку двох змінних аналогічні формули отримати самостійно.

Для зручності розв'язок СЛАР для лінійного і квадратичного наближень доцільно оформити як дві окремі процедури Kramer2, Kramer3.

Для оцінки отриманих наближень застосуємо теоретичне кореляційне відношення

$$\eta^2 = \frac{\sum_{k=1}^{n}\left[f(x_k) - m_f\right]^2}{\sum_{k=1}^{n}(y_k - m_f)^2}, \qquad m_f = \frac{1}{n}\sum_{k=1}^{n} y_k .$$

Слід зауважити що чисельник m_f вже обчислено при формуванні СЛАР і він має позначення b_0. Тобто $m_f = b_0 / n$.

РОЗДІЛ 3 ЧИСЕЛЬНЕ ДИФЕРЕНЦІЮВАННЯ

Часто виникає необхідність у визначенні похідних функції яка задана у послідовності дискретно розташованих точок, тобто у вигляді таблиці або графічно. Для отримання формул чисельного диференціювання цього будується інтерполяційний поліном по декілька сусідніх вузлах і одержану залежність диференціюють. Крім безпосереднього обчислення похідних формули чисельного диференціювання широко застосовуються при чисельному розв'язку диференціальних рівнянь, як звичайних так і з частковими похідними, де вони дозволяють від диференціальних залежностей перейти до алгебраїчних.

Найпростішими, і тому найчастіше вживаними, є формули отримані для рівномірно розташованих вузлів. Так для квадратичної інтерполяції у формі Ньютона по точках x_{i-1}, x_i, x_{i+1}, розташованих на відстані h маємо многочлен

$$f(x) = f(x_{i-1}) + f(x_{i-1}; x_i)(x - x_{i-1}) + f(x_{i-1}; x_i; x_{i+1})(x - x_{i-1})(x - x_i),$$

який за допомогою таблиці поділених різниць приводиться до форми

$$f(x) = f_{i-1} + \frac{f_i - f_{i-1}}{h}(x - x_{i-1}) + \frac{f_{i+1} - 2f_i + f_{i-1}}{2h^2})(x - x_{i-1})(x - x_i) \qquad (3.1)$$

Звідси одержуємо формулу для першої похідної

$$f'(x) = \frac{f_i - f_{i-1}}{h} + \frac{f_{i+1} - 2f_i + f_{i-1}}{2h^2}(2x - x_i - x_{i-1}), \qquad (3.2)$$

Отримана формула може бути використана для будь яких значень $x \in [x_{i-1}; x_{i+1}]$. Так для лівої і правої точок інтервалу відповідно маємо

$$f'(x_{i-1}) = \frac{1}{2h}(-3f_{i-1} + 4f_i - f_{i+1})$$

(3.3)

$$f'(x_{i+1}) = \frac{1}{2h}(f_{i-1} - 4f_i + 3f_{i+1})$$

Але найчастіше частіше вона використовується для обчислення похідної в середній точці, тобто при $x = x_i$

$$f_i' = \frac{f_{i+1} - f_{i-1}}{2h} \qquad (3.4)$$

Диференціювання виразу (3.2) дає формулу для другої похідної

$$f''(x_i) = f_i'' = \frac{f_{i+1} - 2f_i + f_{i-1}}{h^2} \qquad (3.5)$$

Формули (3.4) і (3.5) (вони ще мають назву центральні різниці) за побудовою є точними для многочлена другого ступеня. Щоб отримати кількісну оцінку похибки обчислимо значення $f(x_{i-1})$ і $f(x_{i+1})$ за допомогою розкладення $f(x)$ в ряд Тейлора в околі точки x_i

$$f(x_{i-1}) = f(x_i - h) = f(x_i) - hf'(x_i) + \frac{1}{2!}h^2 f''(x_i) -$$
$$- \frac{1}{3!}h^3 f'''(x_i) + \frac{1}{4!}h^4 f^{IV}(x_i) + \ldots$$

$$f(x_{i+1}) = f(x_i + h) = f(x_i) + hf'(x_i) + \frac{1}{2!}h^2 f''(x_i) +$$
$$+ \frac{1}{3!}h^3 f'''(x_i) + \frac{1}{4!}h^4 f^{IV}(x_i) + \ldots$$

Звідси маємо

$$f_i' = \frac{f_{i+1} - f_{i-1}}{2h} - \frac{h^2}{6} f'''(x_i)$$

$$f_i'' = \frac{f_{i+1} - 2f_i + f_{i-1}}{h^2} - \frac{h^2}{12} f^{IV}(x_i),$$

тобто формула першої похідної (3.4) є як слід було очікувати точною для поліному другого ступеня, а друга центральна різниця - для поліному третього ступеня, і має «додатковий» ступінь точності.

Постільки для обчислення першої похідної у крайніх точках таблиці неможливо використати формулу (3.3), то для цього випадку застосовують менш точні формули одержані шляхом лінійної інтерполяції. Так інтерполяція у формі

$$f(x) = f_{i-1} + \frac{f_i - f_{i-1}}{h}(x - x_{i-1}) \qquad (3.5)$$

приводить до формули лівої різниці

$$f_i' = \frac{f_i - f_{i-1}}{h}, \qquad (3.6),$$

а у формі $f(x) = f_i + \frac{f_{i+1} - f_i}{h}(x - x_i)$ до формули правої різниці

$$f_i' = \frac{f_{i+1} - f_i}{h} \qquad (3.7)$$

Для формул (3.6) і (3.7) величина похибки має порядок $O(h^2)$ і вони є точними тільки для лінійних функцій..

Щоб обчислити похідні вищих порядків можна скористуватися з формул (3.4) і (3.5). Так похідні третього та четвертого порядків відповідно визначаються як

$$f_i''' = \frac{1}{2h}(f''_{i+1} - f''_{i-1}) = \frac{1}{2h^3}(f_{i+2} - 2f_{i+1} + 2f_{i-1} - f_{i-2})$$

$$f_i^{IV} = \frac{1}{h^2}(f''_{i+1} - 2f''_i + f''_{i-1}) = \frac{1}{h^4}(f_{i+2} - 4f_{i+1} + +6f_i - 4f_{i-1} - f_{i-2})$$

Точність наведених формул визначається величиною кроку і тим вища чим менше h . Для практичних цілей достатньо щоб $h \le 0{,}1$.

Операція чисельного диференціювання є *нестійкою* і в тому разі, якщо таблиця одержана в результаті експерименту з великою похибкою то необхідно спочатку наблизити залежність по методу найменших квадратів, а потім обчислити похідні. Такий підхід має назву диференціювання після згладжування.

РОЗДІЛ 4 ЧИСЕЛЬНЕ ІНТЕГРУВАННЯ

4.1 Квадратурні формули типа Н'ютона-Котеса

Формули для наближеного обчислення інтегралів $\int_a^b f(x)dx$ називаються квадратурними. Ця задача виникає у тому випадку коли $f(x)$ завдана у вигляді таблиці, або ж коли аналітичне інтегрування неможливе. В обчислювальній практиці для вирішення цієї задачі найбільш поширеним є підхід коли підінтегральна функція наближено замінюється інтерполяційним поліномом, який легко інтегрується.

Нехай $[a;b]$ - скінчений інтервал осі x, розбитий на n під інтервалів,що називаються елементарними відрізками $[x_i; x_{i+1}]$, $i = 1.2,...,n$. При цьому $x_1 = a, x_{n+1} = b, x_1 = a, x_i < x_{i+1}$. Через $h_i = x_{i+1} - x_i$ позначимо довжину i-го елементарного відрізка. У формулі прямокутників підінтегральна функція наближається константою, в якості якої може бути прийняте будь яке її значення на під інтервалі. Але найточнішого результату можна досягти, коли її обчислити в середній точці $\bar{x}_i = \frac{x_i + x_{i+1}}{2}$. Тоді відповідно для інтервалу і всього відрізка маємо

$$I_i = \int_{x_i}^{x_{i+1}} f(x)dx = h_i f(\bar{x}_i)$$

$$R(f) = \sum_{i=1}^{n} h_i f(\bar{x}_i)$$

Наступні дві формули отримаємо для випадку рівновіддалених вузлів інтегрування , тобто коли $x_{i+1} - x_i = h_i$.

Так інтегрування лінійного наближення

у формі (3.5) приводить до формули

$$\int_{x_{i-1}}^{x_{i+1}} f(x)dx = \frac{h}{2}(f_{i-1} + f_i) ,$$

яка називається формулою трапеції, бо величина інтегралу дорівнює площі трапеції (рис. 4). Після під сумування елементарних інтегралів по всіх інтервалах, на які розбито відрізок $[a,b]$, маємо формулу

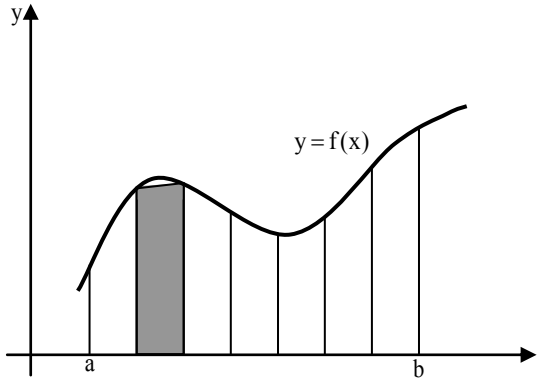


Рис 4

$$T(f) = \int_a^b f(x)dx \approx \frac{h}{2}(f_0 + 2f_1 + 2f_2 + \ldots + 2f_{n-1} + f_n) \quad h = \frac{b-a}{n} \qquad (4.1.1)$$

Наближення підінтегральної функції інтерполяційним поліномом другого ступеня (2.1) дає квадратурну формулу Сімпсона (вона ще має назву формули метода парабол). Відповідно для інтервалу (x_{i-1}, x_{i+1}) і всього відрізка маємо

$$\int_{x_{i-1}}^{x_{i+1}} f(x)dx = \frac{h}{3}(f_{i-1} + 4f_i + f_{i+1})$$

$$S(f) = \int_a^b f(x)dx \approx \frac{h}{3}(f_0 + 4f_1 + 2f_2 + 4f_3 + \ldots + 4f_{n-1} + f_n) \qquad (4.1.2)$$

Якщо у формулі трапецій кількість вузлів може бути довільною, то формулу Сімпсона потребує розбиття відрізка інтегрування на парне число підінтервалів, або ж непарне число вузлів. Використовуючи поліноми більш високих ступенів, можна одержати інші формули для обчислення визначених інтегралів. Квадратурні формули такого типу мають назву формул Н'ьютона-Котеса.

Оцінка похибки квадратурних формул. Формула прямокутників побудована на основі кусково - постійної інтерполяції в той час як формула трапецій на кусково - лінійній інтерполяції. Можна було б очікувати, що формула трапеції буде точніше ніж формула прямокутників. Наприклад, нехай є тільки один

елементарний відрізок $[0,1], n=1$ і $f(x)=x$. Очевидно, формула трапецій дає точний результат, оскільки лінійна функція, що інтерполює $f(x)$ в будь яких двох точках, співпадає з $f(x)$ на всьому інтервалі. В той же час формула прямокутників також дає точний результат, не дивлячись на те, що функція-константа, яка інтерполює $f(x)$ при $x=\frac{1}{2}$ не співпадає з підінтегральною функцією в жодній іншій точці. Середня ж похибка на інтервалі $[0,1]$ дорівнює нулю. Для того, щоб зрозуміти чи є цей приклад типовим, чи формулі прямокутників випадково повезло, треба провезти більш детальний аналіз.

Розкладення в ряд Тейлора для підінтегральної функції $f(x)$ відносно центра цього елементарного відрізка $\bar{x}_i = \frac{x_i + x_{i+1}}{2}$ має вид

$$f(x) = f(\bar{x}_i) + (x-\bar{x}_i) f'(\bar{x}_i) + \frac{1}{2}(x-\bar{x}_i)^2 f''(\bar{x}_i) + \frac{1}{3!}(x-\bar{x}_i)^3 f'''(\bar{x}_i) + \\ + \frac{1}{4!}(x-\bar{x}_i)^4 f^{IV}(\bar{x}_i) + \dots$$

При інтегруванні цього ряду по $[x_i, x_{i+1}]$, відмітимо що

$$\int_{x_i}^{x_{i+1}} (x-\bar{x}_i)^p dx = \begin{cases} h_i, p=0 \\ h_i, p=1 \\ \frac{h_i^2}{12}, p=2 \\ 0, p=3 \\ \frac{h_i^2}{80}, p=4 \end{cases}$$

і інтеграл непарних ступенів дорівнюють нулю. Таким чином

$$\int_{x_i}^{x_{i+1}} f(x)dx = h_i f(\bar{x}_i) + \frac{1}{24} h_i^3 f''(\bar{x}_i) + \frac{1}{1920} h_i^5 f^{IV}(\bar{x}_i) + \ldots (\ )$$

Так як перший додаток в правій частині ( ) є формулою прямокутників, то величина головного члену похибки методу прямокутників складає $\frac{1}{24} h_i^2 f''(\bar{x}_i)$. Повернувшись до ряду Тейлора після підстановки значень $x = x_i$ і $x = x_{i+1}$ маємо

$$f(x_i) = f(\bar{x}_i) - \frac{1}{2} h_i f'(\bar{x}_i) + \frac{1}{8} h_i^2 f''(\bar{x}_i) - \frac{1}{48} h_i^3 f'''(\bar{x}_i) + \\ + \frac{1}{384} h_i^4 f^{IV}(\bar{x}_i) + \ldots$$

$$f(x_{i+1}) = f(\bar{x}_i) + \frac{1}{2} h_i f'(\bar{x}_i) + \frac{1}{8} h_i^2 f''(\bar{x}_i) + \frac{1}{48} h_i^3 f'''(\bar{x}_i) + \\ + \frac{1}{384} h_i^4 f^{IV}(\bar{x}_i) + \ldots$$

Звідки знаходимо

$$\frac{f(x_i) + f(x_{i+1})}{2} = f(\bar{x}_i) + \frac{1}{8} h_i^2 f''(\bar{x}_i) + \frac{1}{384} h_i^4 f^{IV}(\bar{x}_i) + \ldots$$

Поєднуючи останнє співвідношення з розкладенням інтегралу , маємо

$$\int_{x_i}^{x_{i+1}} f(x)dx = h_i \frac{f(x_i) + f(x_{i+1})}{2} - \frac{1}{12} h_i^3 f''(\bar{x}_i) - \frac{1}{480} h_i^5 f^{IV}(\bar{x}_i) + \ldots$$

Звідси витікає, що при малих значеннях h_i головний член похибки методу трапецій на елементарному відрізку складає $-\frac{1}{12} h_i^2 f''(\bar{x}_i)$.

Позначимо $E = \frac{1}{24}\sum_{i=1}^{n} h_i^3 f''(\bar{x}_i)$ $F = \frac{1}{2190}\sum_{i=1}^{n} h_i^5 f^{IV}(\bar{x}_i)$, отримані раніше квадратурні формули прямокутників і трапецій відповідно як $R(f)$ і $T(f)$. Тоді загальна похибка цих методів складає

$$I(f) - R(f) = E + F + \ldots$$

і

$$I(f) - T(f) = -2E - 4F + \ldots.$$

Можна показати, що квадратурна формула Сімпсона в цих позначеннях набуде виду

$$S(f) = \frac{2}{3}R(f) + \frac{1}{3}T(f).$$

Тоді похибку формули Сімпсона можна безпосередньо отримати з виразів для похибок формул прямокутників і трапецій:

$$I(f) - S(f) = \frac{2}{3}\left[I(f) - R(f)\right] + \frac{1}{3}\left[I(f) - R(f)\right] = -\frac{2}{3}F$$

Слід відмітити, що хоча $S(f)$ основана на інтерполяції ступеня два, в вираз для похибки входить четверта похідна і, формула Сімпсона буде точною для кубічних функцій. Тобто, як і формула прямокутників, формула Сімпсона отримує один «додатковий» ступінь точності.

Ми отримали, що головний член похибки формул трапеції і Сімпсона з постійним кроком відповідно дорівнює

$$-\frac{1}{12}h_i^2 f''(\bar{x}_i) \text{ і } \frac{1}{2880}h_i^5 f^{IV}(\bar{x}_i).$$

У випадку формул більш високого порядку точності можна отримати представлення головного члену похибки через похідні вищих порядків. Безпосереднє використання цих виразів для оцінки величини головного члену

похибки є не зовсім зручним, так як потребує виконання операцію диференціювання. В інших задачах вираз для головного члену похибки може бути наскільки складним, що його обчислення потребує додаткового чисельного інтегрування. Тому в обчислювальній практиці застосовується спосіб практичної оцінки похибки, що не використовує фактичного виразу головного члену похибки, опирається лише на факт існування такого члену. Для найпростіших задач цей спосіб називається правилом Рунге і базується на виділенні головного члену похибки за результатом обчислень з двома різними кроками.

Запишемо вираз для головного члену похибки у формі $\delta \approx Ch^q$, (4.1.3)

де q - константа, яка залежить від конкретного методу інтегрування. Щоб виключити невідому константу C обчислимо інтеграл при двох різних кроках - h_1 і h_2. В результаті маємо два наближених значення інтегралу $S_1(h_1)$ і $S_2(h_2)$, Нехай I точне значення інтегралу тоді

$$S_1(h_1)I \approx I + Ch_1^q$$

$$S_2(h_2)I \approx I + Ch_2^q$$

звідки можна знайти константу $C = \dfrac{S_1 - S_2}{h_1^q - h_2^q}$ і обчислити похибку для вибраних величин кроку. Так для h_2 маємо

$$\delta(h_2) = h_2^q \frac{S_1 - S_2}{h_1^q - h_2^q}.$$

Нехай $h_1 = h$, $h_2 = \alpha h, \alpha < 1$. Тоді остання формула набуде більш зручної для обчислень форми

$$\delta(h_2) = \alpha^q \frac{S_1 - S_2}{1 - \alpha^q}.$$

Описана процедура практичної оцінки похибки інтегрування називається першим правилом Рунге .

4.2 Квадратурні формули Гауса і Чебишева

Легко бачити, що формули (4.1.1) і (4.1.2) мають однакову структуру, а саме

$$I = \int_a^b f(x)dx \approx \sum_{k=0}^{N} \omega_k f(x_k), \qquad (4.2.1)$$

де ω_k - вагові коефіцієнти, $f(x_k)$ значення підінтегральної функції у вузлах, а $.n$ - кількість смуг на які розбита область інтегрування. Розташування вузлів і значення вагових коефіцієнтів можна підібрати таким чином, щоб побудована формула була точною для полінома заданого ступеня. Тобто у (4.2.1) параметри ω_i, x_i розглядаються як невідомі, які треба визначити. Нехай будується квадратурна формула по n вузлам, Тоді для визначення координат x_i і вагових коефіцієнтів ω_i маємо $2n$ умови, що дозволяє визначити поліном $2n-1$ ступеня.

Процедуру побудови формул такого типу розглянемо для випадку, коли формула (4.2.1) включає два вузли інтегрування і повинна бути точною для поліному степеня не нижче ніж три. Якщо вибрати для зручності інтегрування відрізок $[-1,1]$, то одержимо наступну систему рівнянь для визначення $x_0, x_1, \omega_0, \omega_1,$

$$f(x) = x^0 \qquad \omega_0 + \omega_1 = \int_{-1}^{1} dx = 2$$

$$f(x) = x^1 \qquad \omega_0 x_0 + \omega_1 x_1 = \int_{-1}^{1} x dx = 0$$

$$f(x) = x^2 \qquad \omega_0 x_0^2 + \omega_1 x_1^2 = \int_{-1}^{1} x^2 dx = \frac{2}{3}$$

$$f(x) = x^3 \qquad \omega_0 x_0^3 + \omega_1 x_1^3 = \int_{-1}^{1} x^3 dx = 0$$

В цих рівняннях ліва частина обчислюється за формулою (4.2.1), а права - точно.

Ця система є нелінійною відносно вузлів і лінійною відносно вагових коефіцієнтів і її рішення є складною задачею навіть для відносно невеликих n. Але той факт, що вузли є нулями многочленів Лежандра ступеня n дозволяє звести задачу до розв'язку СЛАР. Для нашого випадку маємо

$$P_2(x) = \frac{1}{2}(3x^2 - 1),$$

$$x_{1,2} = \mp\frac{\sqrt{3}}{3} \quad \text{і} \quad \omega_1 = \omega_2 = 1.$$

Частковим випадком формули Гауса є квадратурна формула Чебишева, у якій всі вагові коефіцієнти рівні між собою. У цій ситуації маємо $n+2$ невідомі і необхідно брати поліном $n+1$ ступеня. Таким чином квадратурна формула Гауса є точною для поліному ступеня не нижче ніж $2n-1$, а відповідно формула Чебишева не нижче ніж $n+1$.

Перехід від відрізка $(-1;1)$ до довільного відрізка інтегрування $(a;\mathrm{b})$ відбувається шляхом лінійного перетворення $(-1;1)$ в $(a;\mathrm{b})$

$$x_k = \frac{1}{2}\left[(b-a)\xi_k + b + a\right], \quad \text{де} \quad \xi_k \in (-1,1) \qquad (4.2.2)$$

Таким чином квадратурні формули Гауса і Чебишева для довільного відрізка $[a;\mathrm{b}]$ мають відповідно вигляд

$$\int_a^b f(x)dx \approx \frac{b-a}{2}\sum_{k=0}^{n}\omega_k f(x_k)$$

$$\int_a^b f(x)dx \approx \frac{b-a}{n}\sum_{k=1}^{n} f(x_k),$$

де положення вузлів x_k визначаються за формулою (4.2.2).

Значення вузлів інтегрування і вагових коефіцієнтів для квадратурних формул Гауса і Чебишева наведені у всіх підручниках та довідниках по чисельних методах.

Коли підінтегральна функція має вигляд $f(x) = \varphi(x)\sin\omega x$ і на (a,b) для числа коренів $\sin\omega x$ справедлива оцінка $n = \frac{\omega(b-a)}{\pi} >> 1$, то така функція

називається *швидко осцилюючою*. У цьому випадку використання наведених вище квадратурних форму може привести до значної похибки, тому що множник $\max\left|f^{(n+1)}\right| = \omega^{n+1}$ у (4.1.3) є досить великим . Для обчислення інтегралів від швидко осцилюючих функцій використовується метод Файлона. У цьому підході не осцилюючий множник $\varphi(x)$ наближається інтерполяційним поліномом і обчислення інтегралу зводиться до застосування правила інтегрування по частинам. Так у випадку лінійної інтерполяції маємо для інтервалу $(x_{i-1,}x_i)$,

$$I_i = \int_{x_{i-1}}^{x_i} f(x)dx = \int_{x_{i-1}}^{x_i} \left[\varphi_{i-1} + \frac{\varphi_i - \varphi_{i-1}}{h}(x - x_{i-1})\right] \sin \omega x dx =$$

$$= \frac{1}{\omega}\left[\varphi_{i-1}\cos\omega x_{i-1} - \varphi_i \cos\omega x_i + \frac{\varphi_i - \varphi_{i-1}}{\omega h}(\sin\omega x_i - \sin\omega x_{i-1})\right]$$

Якщо відрізок інтегрування $(a;b)$ розбито n точками на $n-1$, інтервалів кожний з яких визначається точками $x_{i-1,}x_i$, то остаточна формула набуває вигляду

4.1 ЛАБОРАТОРНА РОБОТА № 4 ЧИСЕЛЬНЕ ІНТЕГРУВАННЯ

Завдання до лабораторної роботи. Обчислити визначений інтеграл від заданої функції $f(x)$ на вказаному проміжку $[a,b]$ за допомогою формули трапеції, формули Сімпсона і квадратурної формули Гауса

Методичні вказівки до виконання лабораторної роботи.

Як квадратурна формула трапецій

$$I \approx \frac{h}{2}(f_0 + 2f_1 + 2f_2 + ... + 2f_{n-1} + f_n), \qquad (2.6)$$

так і квадратурна формула Сімпсона

$$I \approx \frac{h}{3}(f_0 + 4f_1 + 2f_2 + 4f_3 + \ldots + 4f_{n-1} + f_n) \qquad (2.7)$$

в обчислювальному аспекті є сумами, в які значення функцій у вузлах входять з деякими ваговими коефіцієнтами. Після визначення положення вузлів, наприклад по $i = 0,1,\ldots,n$ формулі $x_i = a + ih$, де n – число вузлів, необхідно обчислити значення в них підінтегральної функції і записати їх у одновимірний масив. Це можна зробити в межах одного циклу. Для обчислення (2.6) початковому значенню суми s присвоїти значення $f_0 + f_n$. Після цього у циклі, параметр якого змінюються від 1 до $n-1$ підсумувати, додатки що лишилися (з ваговим коефіцієнтом 2) і результат помножити на загальний множник.

Легко бачити, що у формулі Сімпсона за винятком першого і останнього додатка, значення функції у вузлах з парними і непарними номерами входять з однаковими коефіцієнтами, але будувати обчислення, виходячи з цієї властивості не зовсім зручно. Більш простим буде початковому значенню суми s присвоїти значення $f_0 + 4f_{n-1} + f_n$ і побудувати цикл типу Repeat – Until, у якому до суми на кожному кроці додається величина $4f_i + 2f_{i+1}$, а параметр циклу збільшується на 2.

Обчислення інтеграла за квадратурною формулою Гауса

$$I \approx \sum_{k=0}^{N} \omega_k f(x_k) \qquad (2.8)$$

є дуже зручним в алгоритмічному плані, тому що всі необхідні дії виконуються в межах одного циклу. Після присвоєння змінній для суми значення нуля, будується цикл, в якому

– читається з файлу поточна координата вузла в інтервалі [-1;1] і відповідний ваговий коефіцієнт;

– за формулою $x_k = \frac{1}{2}\left[(b-a)\xi_k + b + a\right]$ обчислюються координати вузла для даного інтервалу інтегрування;

– в поточному вузлі обчислюється значення підінтегральної функції;

– це значення з поточним ваговим коефіцієнтом додається до суми.

Контрольні питання.

Яким чином можна перевірити правильність обчислення інтегралу?

Чи існують квадратурні формули трапеції або Сімпсона для нерівномірно розташованих вузлів?

Які вихідні дані окрім проміжку інтегрування, підінтегральної функції і числа вузлів інтегрування необхідні для застосування квадратурної формули Гауса?

Як впливає на точність обчислення інтегралу число вузлів інтегрування?

Чи існують якісь обмеження на число вузлів інтегрування?

Яка з квадратурних формул трапеції чи Сімпсона є більш точною і чому?

Коли квадратурні формул трапеції, Сімпсона і Гауса можуть бути застосовані для підінтегральної функції, заданої у вигляді таблиці?

Що є спільним в отриманні квадратурних формул трапеції і Сімпсона?

Вузли інтегрування і вагові коефіцієнти
для квадратурної формули Гауса п’ятого порядку

Номер	1	2	3	4	5
ξ_k	-0,906180	-0,538469	0	0,538469	0,906180
ω_k	0,236927	0,478629	0,568889	0,478629	0,236927

Варіанти завдань до лабораторної роботи.

Варіант	Підінтегральна функція	Інтервал інтегрування
1	$f(x)=x\sin x+\cos x^2$	$-1\le x\le 1$
2	$f(x)=x\cos x+\ln x^2$	$1\le x\le 3$
3	$f(x)=\sin x/x+\ln x$	$1\le x\le 3$
4	$f(x)=x\sin x+\cos x^2$	$0\le x\le \pi$

5	$f(x)=x\sin\sqrt{x}$	$1\le x\le 3$
6	$f(x)=\sqrt{x}\cos x$	$1\le x\le 3$
7	$f(x)=\sin(x^2+1)$	$-2\le x\le 2$
8	$f(x)=2\sin x/(x^2+1)$	$-2\le x\le 2$
9	$f(x)=\ln(x+2)/(x+1)$	$0\le x\le 1$
10	$f(x)=2x^2+\ln(x+1)$	$0\le x\le 2$
11	$f(x)=1/x^2-e^{-x}$	$1\le x\le 3$
12	$f(x)=\sin x/x-x$	$1\le x\le 2$
13	$f(x)=x^3\cos(x-1)$	$-2\le x\le 1$
14	$f(x)=\sqrt{x}e^{-x}+\cos x$	$-1\le x\le 1$
15	$f(x)=\sqrt{x}\ln x+x$	$1\le x\le 3$
16	$f(x)=\sqrt{x}\sin x$	$1\le x\le 3$
17	$f(x)=x^2\cos x$	$-1\le x\le 1$
18	$f(x)=1/\sqrt{x+\cos x}$	$1\le x\le 3$
19	$f(x)=2\cos x/\left(x^2+1\right)$	$-1\le x\le 1$
20	$f(x)=\ln\sqrt{x}/(x+1)$	$1\le x\le 3$
21	$f(x)=\cos(x^2+1)/x$	$1\le x\le 3$
22	$f(x)=\sin\left(\cos x\right)$	$-\pi\le x\le\pi$
23	$f(x)=x^2\cos(x^2+1)$	$-1\le x\le 1$
24	$f(x)=\ln\sqrt{x}+\cos x$	$1\le x\le 3$
25	$f(x)=\cos(x^2+1)/(x^2+1)$	$-\pi\le x\le\pi$

РОЗДІЛ 5 ЧИСЕЛЬНІ МЕТОДИ АЛГЕБРИ

До задачі цього класу належать задачі лінійної алгебри і розв’язок нелінійних рівнянь і систем.

Прийнято виділяти чотири основні задачі лінійної алгебри: розв'язок систем лінійних алгебраїчних рівнянь $AX = B$; обчислення визначників $\det(A)$; знаходження оберненої матриці A^{-1}; визначення власних значень і векторів. Так як центральне місце у цих задачах займають матриці нагадаємо деякі основні їх властивості.

Типи матриць. Квадратна матриця A порядку n складається з n^2 елементів a_{ij}. Якщо тільки деякі елементи a_{ij} відрізняються від нуля, матриця називається розрідженою. Ясно, що при відповідному кодуванні її можна представити набагато меншім ніж n^2 числом, так як відпадає необхідність зберігати нульові елементи. Матриця, більшість елементів якої відрізняється від нуля, називається щільною матрицею. Термін щільність використовується для позначення відношення ненульових елементів до n^2.

Іноді, навіть якщо жоден з елементів не дорівнює нулю, елементи a_{ij} можуть бути обчислені по індексам i, j за допомогою простих формул. В цьому випадку матриця називається такою, що породжується і її елементи не потребують зберігання. Якщо, навпаки, елементи матриці представлені як n^2 чисел, то матриця називається такою, що зберігається. При цьому не важливо чи є деякі елементи нулями, чи ні, так як нулі теж необхідно пам'ятати..

Матриця A називається стрічковою, якщо $a_{ij} = 0$ для всіх $|i - j > m|$, тому що ненульові елементи утворюють смугу або стрічку, вздовж головної діагоналі. При цьому число діагоналей в смузі дорівнює $2m + 1$.

Алгоритми для розв'язку різних обчислювальних задач ,лінійної алгебри відрізняються відповідного до того, змінюють вони в процесі розв'язку елементи матриці, чи ні. Найбільш ефективні методи для матриць невисокого порядку, що зберігаються, як правило змінюють матрицю. Разом з цим такі методи, застосовані до розріджених матриць, в процесі обчислень збільшують щільність матриці. Отже, якщо матриця має високий порядок і є рідкою, то в

загальному випадку неможливо використати методи найбільш придатні для невеликих матриць, що зберігаються

Розв'язок систем лінійних алгебраїчних рівнянь, є мабуть, найбільш поширеною у обчислювальній практиці задачею. Найчастіше джерелом систем лінійних рівнянь є апроксимація неперервного функціонального рівняння скінченно-різницевою задачею. Наприклад задачу Діріхле для оператора Лапласа (див. розділ VII) можна апроксимувати великою системою простих скінченно-різницевих рівнянь. Матриці, пов'язані з різницевими рівняннями , майже завжди є великими і розрідженими. Друге дуже суттєве джерело систем лінійних рівнянь - розв'язок лінійних задач методом найменших квадратів. Як правило матриці у цьому випадку є відносно невеликими і щільними.

Частіше всього обчислювальні задачі лінійної алгебри розв'язуються точними і ітераційними методами. Метод відноситься до класу точних, якщо в допущенні про відсутність округлень він дає точне рішення після скінченного числа арифметичних і логічних операцій. В терміні точний є суттєва частка ідеалізації, так як при реальних обчисленнях похибок округлення уникнути не вдається. Тому далі для цих методів буде вживатися термін прямі методи. Метод називається ітераційним, якщо розв'язок отримано у вигляді границі елементів деякої послідовності.

Якщо число ненульових елементів матриці має порядок n^2, то для більшості точних методів,що використовуються в даний час для розв'язку таких систем необхідне число операцій має порядок n^3. Більшість точних методів відносяться до так званих методів виключення.

5.1 Прямі методи розвязк СЛАР

Найбільш поширеними формами запису СЛАР є по компонентна $\sum_{k=1}^{n} a_{mk} x_k = b_m$ і матрична $AX = B$.

Метод виключення Гауса належить до прямих (точних) методів і є найбільш універсальним і поширеним методом рішення задач цього класу,що і використовуються для систем з матрицями довільної структури. В результаті застосування методу матриця системи набуває трикутної форми

$$\begin{aligned}
a_{11}x_1 + a_{12}x_2 + a_{13}x_3 + \ldots + a_{1n-1}x_{n-1} + a_{1n}x_n &= b_1 \\
a_{22}x_2 + a_{23}x_3 + \ldots + a_{2n-1}x_{n-1} + a_{2n}x_n &= b_2 \\
\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots & \qquad (5.1.1) \\
a_{n-1n-1}x_{n-1} + a_{n-1n}x_n &= b_{n-1} \\
a_{nn}x_n &= b_n
\end{aligned}$$

Тобто останнє рівняння містить тільки одну невідому, яку можна знайти у явному вигляді, потім підставити у передостаннє рівняння і т.д. Якщо з першого рівняння виключити невідому x_1 то коефіцієнти матриці і компоненти вектора правої частини після приведення подібних набудуть вигляду

$$a_{ij} = a_{ij} - \frac{a_{1j}}{a_{11}} a_{i1} \quad b_i = b_i - \frac{b_1}{a_{11}} a_{1i} \quad i = 2,3,\ldots,n$$

Повторивши цю операцію до x_{n-1} включно отримуємо, що для зведення матриці до вигляду (5.1.1) потрібно перерахувати її елементи і компоненти вектора правої частини за формулами

$$a_{ij} = a_{ij} - \frac{a_{kj}}{a_{kk}} a_{ik} \qquad b_i = b_i - \frac{b_k}{a_{kk}} a_{ki} \qquad (5.1.2)$$

Ця частина алгоритму називається прямою ходою метода. Коефіцієнт a_{kk} у (5.1.2) називаєтьсь провідним елементом і в тому випадку, коли на якомусь кроці виключення його величина по модулю близька до нуля, необхідно поміняти місцями рядки матриці з номерами k і q, $\mathrm{q} > \mathrm{k}$, щоб виконувалась умова $|a_{kk}| \geq \varepsilon$. Якщо ж жодна з таких перестановок не дає потрібного результату, то це значить, що матриця системи є виродженою і система рішень не має.

На оберненій ході визначаються власне компоненти вектора рішення

$$x_n = \frac{b_n}{a_{nn}} \quad , \quad x_k = \frac{1}{a_{kk}} \left(b_k - \sum_{j=k+1}^{n} a_{kj} x_j \right)$$

Слід відзначити, що ідея виключень Гауса використовується не тільки для розв'язку СЛАР. Цей підхід є дуже ефективним при обчислення визначників та обернених матриць.

Після приведення матриці до трикутного вигляду її визначник може бути обчислено розкриттям по елементам першого стовпця. Так чином отримуємо формулу $\det(A) = \prod_{k=1}^{n} a_{kk}$.

Матриця A^{-1} називається оберненою до матриці A, якщо $AA^{-1} = A^{-1}A = I$, де I одинична матриця.

При розв'язку цієї задачі за допомогою класичної формули лінійної алгебри

$$a_{ij}^{-1} = \frac{A_{ji}}{\det(A)}$$

необхідно обчислити один визначник n -го порядку і для визначення алгебраїчних доповнень A_{ji} - n визначників $(n-1)$ -го порядку.

Нехай X_i, i -1,2,...,n вектори, що отримані в результаті розв'язку n СЛАР $AX_i = B_i$, у яких праві частини визначені наступним чином

$$B_1 = \begin{bmatrix} 1 \\ 0 \\ . \\ . \\ . \\ 0 \end{bmatrix}, \; B_2 = \begin{bmatrix} 0 \\ 1 \\ . \\ . \\ . \\ 0 \end{bmatrix}, \ldots, \; B_n = \begin{bmatrix} 0 \\ 0 \\ . \\ . \\ . \\ 1 \end{bmatrix} \qquad (5.1.3)$$

Якщо тепер утворити матриці M і B , яких стовпцями відповідно будуть вектори X_i і B_i, то $AM = B$. Постільки згідно з (5.1.3) матриця B є одиничною, то маємо $A^{-1} = M$.

Легко бачити, що обсяг обчислень в описаному методі є набагато меншим ніж за цією формулою ( ).

Досить часто матриця системи має спеціальну особливу будову, і застосовувати у цьому разі метод Гауса є недоцільним. Так при рішенні рівнянь з частковими похідним виникають матриці з діагонально- домінуючою будовою. У цьому випадку використовуються методи типу прогонки.

Метод простої прогонки є частковим випадком методу Гауса, для систем з трьохдіагональними матрицям, тобто які мають форму (Рис. 5.1)

$$A=\begin{bmatrix} * & * & 0 & 0 & 0 & 0 & 0 \\ * & * & * & 0 & 0 & 0 & 0 \\ 0 & * & * & * & 0 & 0 & 0 \\ 0 & 0 & * & * & * & 0 & 0 \\ 0 & 0 & 0 & * & * & * & 0 \\ 0 & 0 & 0 & 0 & * & * & * \\ 0 & 0 & 0 & 0 & 0 & * & * \end{bmatrix},$$

Рис. 5.1- Структура трьохдіагональної матриці

а відповідні їм системи записуються у вигляді

$$\begin{cases} B_1x_1 + C_1x_2 = D_1 \\ A_ix_{i-1} + B_ix_i + C_ix_{i+1} = D_i \\ A_nx_{n-1} + B_nx_n = D_n \end{cases} \qquad (5.1.4)$$

Процес рішення тут теж розбивається на два етапи. Якщо припустити, що невідомі x_{i-1}, x_i зв'язані співвідношенням

$$x_{i-1} = R_ix_i + \theta_i \ , \qquad (5.1.5)$$

то з (5.1.4) і (5.1.5) можна отримати рекурентні формули для визначення прогоночних коефіцієнтів R, θ (пряма хода метода

$$R_{i+1} = -\frac{C_i}{A_iR_i + B_i} \ , \qquad \theta_{i+1} = \frac{D_i - A_i\theta_i}{A_iR_i + B_i},$$

$$R_2 = -\frac{C_1}{B_1}; \qquad \theta_2 = \frac{D_1}{B_1},$$

Після цього визначається невідома $x_n = \dfrac{D_n - A_n\theta_n}{A_n R_n + B_n}$.

На оберненій ході за формулою $x_n = x_{i+1}R_{i+1} + \theta_{i+1}$

обчислюються невідомі, що залишилися: $x_{n-1}, x_{n-2}, \ldots, x_1$.

Окрім розглянутого методу виключень Гауса існують і інші способи зведення матриць до трикутного виду. В **методі** LU **- розкладання** (LU - факторізації) вихідна матриця представляється у вигляді добутку $A = LU$, де L - нижня ,а U - верхня трикутні матриці, які мають вид

$$L = \begin{pmatrix} l_{11} & 0 & 0 \ldots\ldots\ldots\ldots 0 \\ l_{21} & l_{22} & 0 \ldots\ldots\ldots\ldots 0 \\ l_{31} & l_{32} & l_{33}\ 0 \ldots\ldots\ldots 0 \\ \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots \\ l_{n1} & l_{n2} & l_{n3} \ldots\ldots\ldots\ldots l_{nn} \end{pmatrix} \qquad U = \begin{pmatrix} u_{11} & u_{12} & u_{13} \ldots\ldots\ldots\ldots u_{1n} \\ 0 & u_{22} & u_{23} \ldots\ldots\ldots\ldots u_{2n} \\ 0 & 0 & u_{33} \ldots\ldots\ldots\ldots u_{3n} \\ \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots \\ 0 & 0 & 0 \ldots\ldots\ldots\ldots u_{nn} \end{pmatrix}$$

і для яких

$$l_{ij} = 0 \text{ при } j > i, \quad u_{ij} = 0 \text{ при } j > i \ u_{ij} = 0 \qquad (5.1.6)$$

Тоді СЛАР набуває вигляду $AX = LUX = B$. Позначивши $Y = UX$,
замість початкової задачі з матрицею загального вигляду, приходимо до двох систем з трикутними матрицями $LY = B$ і $UX = Y$, развязок яких проводиться послідовно.

Так як L і U визначені з точністю до множення U на константу і ділення L на ту ж константу, то можна допустити, щоб $l_{ii} = 1$. Використаємо при визначення елементів LU - розкладення правилом множення матриць

$$a_{ij} = \sum_{k=1}^{n} l_{ik} u_{kj}$$

З рахуванням (5.1.6для першого стовпця $j=1$ маємо $a_{i1}=l_{i1}u_{11}$, звідки при $i=1$ $u_{11}=a_{11}$, а для всіх елементів стовпця $l_{i1}=\dfrac{a_{i1}}{u_{11}}$

Аналогічна дія з компонентами першого рядка дає $a_{1i}=l_{11}u_{1i}$ звідки $u_{1j}=a_{1j}$.

Формули для інших компонент матриць L і U нескладно отримати за індукцією. Для елементів a_{22} і a_{32} згідно з а ( ) можна записати

$$a_{22}=l_{21}u_{12}+l_{22}u_{22},\quad a_{32}=l_{31}u_{12}+l_{32}u_{22}$$

звідки для компонент шуканих матриць маємо

$$u_{22}=a_{22}-l_{21}u_{12}\qquad l_{32}=\frac{1}{u_{22}}(a_{32}-l_{31}u_{12})$$

Дві останні формули легко узагальнити на випадок довільних індексів, що приводить до наступних формул спочатку обчислюються компоненти i-го рядка матриці U

$$u_{ij}=a_{ij}-\sum_{k=1}^{i}l_{ik}u_{kj}\qquad j=i,\ldots,n$$

а потім компоненти i-го стовпця матриці L

$$l_{ji}=\frac{1}{u_{ii}}\left(a_{ij}-\sum_{k=1}^{i}j_{jk}u_{ki}\right)\qquad j=i+1,\ldots,n$$

Повернемося до розв'язку СЛАР $AX=B,$ який як вже було згадано, проводиться в два етапи. На першому кроці з системи $LY=B$ знаходиться вектор Y, компоненти
якого обчислюються як

$$y_1=\frac{b_1}{l_{11}}\quad,\quad x_i=\frac{1}{l_{ii}}\left(b_i-\sum_{k=i-1}^{n}l_{ik}y_k\right),\ i=2,3,..n,$$

після чого з системи $UX = Y$ по формулам оберненої ходи

$$x_n = \frac{y_n}{u_{nn}} \quad , \quad x_i = \frac{1}{u_{ii}}\left(y_i - \sum_{k=i+1}^{n} u_{ik} x_k \right), \; i = n-1, n-2, \ldots 1$$

отримуємо вже остаточний результат, тобто вектор X

Модифікація LU- розкладання у випадку коли матриця A є симетричною носить назву **методу Холецького** або **методу квадратного кореня**. СЛАР з симетричними матрицями досить часто зустрічаються в обчислювальній практиці. Так до таких систем зводиться задача наближення функцій по методу найменших квадратів. Цілком природно використати властивість симетрії матриці для підвищення ефективності алгоритму і тоді обсяг обсяг обчислень можна скоротити майже вдвоє .

Для довільного елементу матриці маємо $a_{ij} = \sum_{k=1}^{n} l_{ik} u_{kj}$

У нашому випадку $U = L^T$ для елементів якої $l_{ij}^T = l_{ij}$ і маємо $A = LL^T$.

Окрім цього при $j > i$ $l_{ij}^T = 0$. З врахування наведених властивостей, маємо

$$a_{ij} = \sum_{k=1}^{n} l_{ik} u_{kj} = \sum_{k=1}^{n} l_{ik} l_{jk} == \sum_{k=1}^{i} l_{ik} l_{jk} = \sum_{k=1}^{i-1} l_{ik} l_{jk} + l_{ii} l_{ji}$$

Якщо в покласти в цій формулі $j = i$, то для діагональних елементів знаходимо

$$l_{ii} = \sqrt{a_{ii} - \sum_{k=1}^{i-1} l_{ik}^2}, \qquad l_{11} = \sqrt{a_{11}} \quad (5.1.7)$$

а для всіх інших елементів i -го стовпця

$$l_{ji} = \frac{1}{l_{ii}}\left(a_{ij} - \sum_{k=1}^{i-1} l_{ik} l_{jk} \right) j = \mathrm{i} + 1, \ldots, \mathrm{n} \quad (5.1.8)$$

Як і у випадку LU - розкладення задача зводиться до послідовного розв'язку двох систем з трикутними матрицями $LY = B$ $L^T X = Y$, але останній крок матиме вид

$$x_n = \frac{y_n}{l_{nn}} \quad x_i = \frac{1}{l_{ii}}\left(y_i - \sum_{k=i+1}^{n} l_{ki} x_k \right), \; i = n-1, n-2, \ldots 1$$

Здійсненню дійсного розкладання симетричної матриці A можуть завадити дві обставини: рівність нулю елемента l_{ii} і від'ємність виразу під коренем. Але відомо, що для важливого у за стосунках класу симетричних додатно визначених матриць розкладення по формулах (5.1.7) (5.1.7), є можливим.

5.2 Ітераційні методи розв'язку СЛАР

Як уже вже відзначалось вище, ітераційні методи є особливо ефективними, коли початкове наближення близьке до дійсного рішення. У випадку СЛАР це відбувається коли кілька разів необхідно розв'язувати системи, у яких елементи матриць відрізняються на малі величини. Крім цього застосування ітераційних методів доцільне у наступних випадках:

- система має настільки високий вимір, що елементи матриці не зберігаються в пам'яті ЕОМ, а обчислюються безпосередньо в процесі розв'язку СЛАР;

- матриця системи має розріджену структуру.

Нехай у першому рівнянні системи всі невідомі окрім x_1 рівні відповідно $x_i = x_i^{(k)}, i = 2,..,n$, тоді

$$x_1 = x_1^{(k+1)} = \frac{1}{a_{11}}(b_1 - a_{12}x_2^{(k)} - a_{13}x_3^{(k)} - a_{1n}x_n^{(k)})$$

Аналогічно з другого рівняння маємо

$$x_2 = x_2^{(k+1)} = \frac{1}{a_{22}}(b_2 - a_{21}x_1^{(k)} - a_{23}x_3^{(k)} - a_{2n}x_n^{(k)})$$

Легко бачити, що обсяг обчислень можна набагато скоротити, якщо попередньо поділити кожний рядок СЛАР на діагональний елемент, і таким чином виключити операцію ділення з ітераційного процесу:

$$\alpha_{ij} = \frac{a_{ij}}{a_{ii}},\ \beta_i = \frac{b_i}{a_{ii}}$$

Таким чином маємо формулу *метода простої ітерації*

$$x_i^{(k+1)} = \beta_i - \sum_{j=1}^{i-1} \alpha_{ij} x_j^{(k)} - \sum_{j=1}^{i+1} \alpha_{ij} x_j^{(k)}$$

Постільки заміна компонент $x^{(k)}$ на $x^{(k+1)}$ відбувається після закінчення ітерації по всіх номерах, То цей метод ще називають методом одночасного зсуву. Його суттєвим недоліком є те, що інформація про нове наближення для компоненти враховується з запізненням. Якщо в уточненні $x_i^{(k+1)}$ врахувати компоненти $x_j^{(k+1)}$, $\mathrm{j} = 1,2,..,\mathrm{i}-1$, отримаємо формулу *методу Зейделя*

$$x_i^{(k+1)} = \beta_i - \sum_{j=1}^{i-1} \alpha_{ij} x_j^{(k+1)} - \sum_{j=1}^{i+1} \alpha_{ij} x_j^{(k)}$$

Для збіжності метода Зейделя достатньо, щоб діагональний елемент у кожному рядку був домінуючим $|a_{ii}| \geq |a_{ij}|,\ \ \mathrm{j} = 1,2,..,\mathrm{n}$

Нехай на $(k+1)$ ітерації знайдено уточнення компоненти x_i^*. Якщо в якості наступного наближення брати не безпосередньо це значення, а визначити його як

$$x_i^{(k+1)} = x_i^{(k)} + \omega(x_i^* - x_i^{(k)}) \ ,0 \leq \omega \leq 2 \qquad (5.2.1)$$

то отримаємо *метод релаксації*. При $\omega > 1$ він називається методом верхньої релаксації, при $\omega < 1$ - методом нижньої релаксації. При $\omega = 1$ отримуємо частковий випадок – метод Зейделя. Величина параметра ω регулює

швидкість збіжності і хоч існують теоретичні підходи для визначення його оптимального значення, але звичайно ω для конкретної матриці знаходиться шляхом чисельного експерименту. Метод релаксації є найбільш ефективним для розв'язку систем високого виміру з розрідженою матрицею. У цьому випадку з матриці виділяється блок і уточнюються ті компоненти вектора X , які входять до блоку. Цей варіант методу називається *блочною релаксацією*.

5.3 Міра обумовленості матриці

Що отримати більш точну і надійну міри близькості матриці до виродженості ніж значення визначника або провідного елементу, розглянемо норму вектора. Найбільш вживаною векторною нормою є евклідова довжина

$$\|X\| = \left(\sum_{i=1}^{n} x_i^2 \right)^{1/2}$$

Але застосування цієї норми , потребує значного обсягу обчислень. Тому у матричній алгебрі застосовується наступна норма

$$\|X\| = \sum_{i=1}^{n} |x_i|,$$

яка має більшість властивостей евклідової довжини. Множення вектора X на матрицю A приводить до нового вектору AX, норма якого може суттєво відрізнятися від норми вектора X. Область можливих змін може бути задана двома числами

$$M = \max \frac{\|AX\|}{\|X\|}, \; m = \min \frac{\|AX\|}{\|X\|},$$

де максимуми і мінімуми беруться по всім ненульовим векторам.

$$cond(A) = \frac{M}{m}$$

Розглянемо систему рівнянь $AX = B$, і іншу систему , одержану шляхом зміни правої частини: $A(X + \Delta X) = B + \Delta B$. Оскільки $A(\Delta X) = \Delta B$ то з визначень M і m негайно випливають нерівності

$$\|B\| \le M\|X\|$$

$$\|\Delta B\| \ge m\|\Delta X\|$$

Якщо першу нерівність в ( )поділити на другу , отримаємо підсилену нерівність

$$\frac{\|B\|}{\|\Delta B\|} \le \frac{M}{m}\frac{\|X\|}{\|\Delta X\|}$$

Ліву і праву частину в останній нерівності помножимо на $\frac{\|\Delta X\|}{\|X\|}\frac{\|\Delta B\|}{\|B\|}$,

і прийдемо до остаточної оцінки мірі обумовленості матриці

$$\frac{\|\Delta X\|}{\|X\|} \le cond(A)\frac{\|\Delta B\|}{\|B\|}$$

Отримана нерівність показує, що число обумовленості виконує роль множника в збільшенні відносної похибки. Число обумовленості можна розглядати як величину обернену до відносної відстані від даної матриці до множини вироджених матриць. Наведемо деякі властивості числа обумовленості. Ясно, що и $M \ge m$ і тому $cond(A) \ge 1$. Якщо A множиться на скаляр c, то і M і m будуть помножені на модуль цього скаляра, так що

$$cond(cA) = cond(A)$$

Якщо D - діагональна матриця,то

$$cond(D) = \frac{\max\|d_{ii}\|}{\min\|d_{ii}\|}$$

Останні дві властивості в певній мірі пояснюють, чому $cond(A)$ є кращою мірою близькості до виродженості , ніж визначник матриці A. В якості граничного прикладу розглянемо діагональну матрицю порядку 100 з числом 0,1 на головній діагоналі, тоді $\det(A) = 10^{-100}$, що звичайно вважається малим числом. Але $cond(A) = 1$і компоненти вектора Ax лише множником 0,1відрізняються від відповідних компонент вектора x. Для лінійних систем така матриця Aповодить себе скоріше як одинична, а не вироджена.

Розглянемо наступний приклад. Для СЛАР з матрицею

$A = \begin{pmatrix} 4{,}1 & 2{,}8 \\ 0{,}7 & 6{,}6 \end{pmatrix}$ і вектором правих частин $B = \begin{pmatrix} 4{,}1 \\ 9{,}7 \end{pmatrix}$ розв'язком є вектор $X = \begin{pmatrix} 1 \\ 0 \end{pmatrix}$, норма якого $\|X\| = 1$. Крім цього $\|B\| = 13{,}8$ $\det(A) = -0{,}1$, тобто величина визначника суттєво відрізняється від нуля. Якщо замінити праву частину на . $B' = \begin{pmatrix} 4{,}11 \\ 9{,}70 \end{pmatrix}$, то розв'язком буде вектор $X' = \begin{pmatrix} 0{,}34 \\ 0{,}97 \end{pmatrix}$.

Нехай $\Delta B = B - B'$, і $\Delta X = X - X'$. Тоді $\|\Delta B\| = 0{,}01$ $\|\Delta X\| = 1{,}63$. Дуже мале збурення внесене в B, зовсім змінило X . Дійсно, відносні зміни дорівнюють

$$\frac{\|\Delta B\|}{\|B\|} = 0{,}0007246,\ \frac{\|\Delta X\|}{\|X\|} = 1{,}63.$$

Так як $cond(A)$ характеризує максимально можливе збільшення, то

$$cond(A) \geq \frac{1{,}63}{0{,}0007246} = 2249.$$

Насправді вибрані величини B і ΔB як раз дають максимум, то для даного прикладу $cond(A) = 2249$.

5.3 Задачі на власні значення

5.4

У багатьох випадках необхідно мати такі важливі характеристики матриці як власні числа та власні вектори. За визначенням, якщо для деякого числа λ та вектора виконується умова

$$AX = \lambda X, \qquad (5.4.1)$$

то λ називається власним числом матриці A, а X - власним вектором. Умова (5.4.1) може бути зображена у вигляді СЛАР

$$(A - \lambda E)x = 0,$$

де E - одинична матриця. Ця система є однорідною і має рішення відмінні від нуля тільки тоді коли

$$\det(A - \lambda E) = 0 .$$

Розкриваючи визначник, отримуємо многочлен n -го ступеня , корені якого і будуть власними значеннями. Тобто матриця розмірності $n \times n$ має n власних чисел, серед яких можуть бути і комплексні навіть для матриць з дійсними елементами. Якщо треба знайти всі власні значення і відповідні їм власні вектори, то така задач називається *повною проблемою власних значень*. На практиці досить часто достатньо знайти найбільше або найменше власне число. В цьому випадку маємо *часткову проблему власних значень*.

На практиці визначення власних значень шляхом пошуку коренів поліному використовується або ж для матриць невисоких вимірів, а в основному для обчислення комплексних λ. Для побудови методів рішення

проблеми власних чисел використовуються наступні властивості власних чисел і векторів:

симетрична матриця елементи якої є дійсними числами має тільки дійсні власні значення;

власні вектори, що відповідають різним власним числам ортогональні.

власні значення подібних матриць A і B рівні між собою а власні вектори зв'язані співвідношенням $X = PY$ (Матриці A і B називаються подібними, якщо існує не вироджена матриця P така, що

$$B = P^{-1}AP,\ \det(\mathrm{P}) \neq 0\ ).$$

Повна проблема власних значень є досить складною та трудомісткою за винятком випадків трикутної та діагональної матриць коли власні значення знаходяться безпосередньо:

$$\det(A - \lambda E) = \prod_{i=1}^{n}(a_{ii} - \lambda_i) = 0\ .\ \ \lambda_i = a_{ii}\ ,\ \mathrm{i} = 1,2,...,\mathrm{n}$$

Якщо ж звести матрицю до трикутної форми за допомогою гаусових виключень, то власні значення результуючої матриці не будуть власними значеннями початкової. Справа у тому, що перетворення виключення не є перетвореннями подібності. Поняття подібних перетворень є основою *методу обертань,* який є самим поширеним для рішення повної проблеми. Ідея цього методу в тому що шляхом подібних перетворень початкова матриця зводиться до діагональної. За матрицю P зручно взяти ортогональну матрицю U, тобто таку, що $U^T U = E$.

Формально схема методу дається формулою

$$A^{(k+1)} = U^T A^{(k)} U \qquad (5.4.2)$$

На кожній ітерації з матриці $A^{(k)}$ вилучається найбільший за модулем недіагональний елемент a_{ij} , Для цього матриця U вибирається так, щоб у неї відмінні від нуля були тільки чотири елементи

$$U_{ii} = \cos\varphi \quad U_{ij} = -\sin\varphi \quad U_{ji} = \sin\varphi \quad U_{jj} = \cos\varphi \qquad (5.4.3)$$

Після підстановки (5.4.3) в (5.4.2 отримуємо рівняння для визначення величини φ

$$a_{ij}^{(k+1)} = (a_{ij}^{(k)} \cos\varphi - a_{ii}^{(k)} \sin\varphi)\cos\varphi + (a_{jj}^{(k)} \cos\varphi - a_{ji}^{(k)} \sin\varphi)\sin\varphi$$

У випадку симетричної матриці значення φ визначається безпосередньо.

$$tg\varphi = 2\frac{a_{ij}}{a_{ii} - a_{jj}}$$

Метод обертань використовується тільки для симетричних дійсних матриць і причиною цього обмеження є не скільки простота одержання параметру φ, а та обставина, що у випадку несиметричних матриць власні числа можуть бути комплексними. Постільки для діагональної матриці $\lambda_i = a_{ii}$, а перетворення (5.4.2) не переводять дійсні числа в комплексні, то спроба використати цей метод для несиметричних матриць може бути не коректною(приведе до розбіжності метода). Теоретично метод потребує безкінечного числа ітерацій, але на практиці збігається досить швидко.

Часткова проблема власних значень досить просто рішається ітераційними методами. Візьмемо довільний вектор $x^{(0)}$ і розкладемо його по базису, утвореному власними векторами матриці A

$$x^{(0)} = \sum_{i=1}^{n} c_i x_i$$

Застосуємо до $x^{(0)}$ n разів матрицю A і з врахуванням визначення (5.3.1) будемо мати

$$x^{(n)} = \sum_{i=1}^{n} \lambda_i^n c_i x_i$$

Нехай власні числа пронумеровані у порядку зменшення $\lambda_1 \geq \lambda_2 \geq \ldots \geq \lambda_{1n}$. Очевидно, що існує такий номер n, починаючи з якого виконується співвідношення

$$x^{(n)} = \lambda_1^n c_1 x_1 + O(|\lambda_2|^n \qquad (5.4.2)$$

Вважаючи, що останній додаток в (5.4.2) є величиною, якою нехтують, одержимо наступний алгоритм для знаходження найбільшого власного числа:

Вибирається початкове наближення для вектора $x^{(0)}$ і обчислюється вектор

$$y_m^{(0)} = \sum_{k=1}^{n} a_{mk} x_k^{(0)}$$

Наступне наближення визначається як $x_m^{(1)} = \frac{1}{\lambda^{(1)}} y_m^{(0)}$

де $\lambda^{(1)}$ найбільша по абсолютній величіні компонента $\lambda_m^{(0)}$. Умовою зупинки алгоритму може бути як досягнення збіжності по власному числу

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

так і по власному вектору

$$\left\|x^{(k+1)} - x^{(k)}\right\| = \left[\sum_{n=1}^{N} (x_n^{(k+1)} - x_n^{(k)})^2\right]^{1/2} \le \varepsilon.$$

Швидкість цього методу, як і всіх ітераційних, у значній мірі визначається початковим наближенням. Але слід відмітити, що у випадку $\frac{\lambda_1}{\lambda_2} \approx 1$, то як витікає з (5.4.2), збіжність методу може суттєво уповільнитися. Щоб знайти найменше власне число цей же алгоритм застосовується до матриці оберненої даній.

5.5 Нелінійні рівняння

В залежності від вигляду функції $f(x)$ нелінійні рівняння

$$f(x)=0$$

поділяються на трансцендентні та алгебраїчні і як буде показано нижче, такий поділ доцільно ввести і для методів їх рішення. Слід відзначити, всі існуючі методи для задач цього класу(за винятком квадратних і кубічних рівнянь) є ітераційними.

В якості кількісної характеристики ефективності ітераційних методів служить швидкість збіжності. Нехай x - корінь рівняння, який необхідно визначити. Звичайно користуються наступними двома визначеннями.

Метод збігається з швидкістю геометричної прогресії зі знаменником q, якщо для всіх $n \in N$ справедлива оцінка

$$|x_{n+1}-x| \le Cq^n |x_n - x|$$

Якщо в деякому околі кореня виконується умова

$$|x_{n+1}-x| \le C|x_n - x|^p, \text{ де } C>0,$$

то число p називають швидкістю або порядком збіжності.

Спочатку розглянемо трансцендентні рівняння.

Метод простої ітерації Попередньо задача $f(x)=0$ приводиться до вигляду

$$x=\varphi(x) \qquad (5.5.1)$$

Тобто в геометричній інтерпретації розв'язок зводиться до визначення точки перетину графіків $y=x$ і $y=\varphi(x)$. Точка x, яка задовольняє умові (5.5.1) називається нерухомою точкою, а функція $\varphi(x)$- стискуючою функцією.

Ітерація $x_n=\varphi(x_{n-1})$, $n=1,2,3,\ldots$ називається ітерацією нерухомої точки . Нехай $\varphi(x)$ неперервна функція, а $\{x_n\}_{n=0}^{\infty}$ - послідовність точок ,що генерується за допомогою ітерацій $x_n=\varphi(x_{n-1})$ Якщо $\lim\limits_{n\to\infty} x_n = x^*$, то x^* є нерухомою точкою $\varphi(x)$.

Розглянемо поведінку послідовності $\{x_n\}_{n=0}^{\infty}$ в околі нерухомої точки. Нехай $\varepsilon_n = x_n - x^*$ - похибка на n -му кроці роботи метода. Якщо праву частину співвідношення $x_{n+1} = \varphi(x_n)$ розкласти в ряд Тейлора, обмежившись тільки лінійною частиною , то для визначення похибки на $n+1$ -му кроці маємо

$$x^* + \varepsilon_{n+1} = \varphi(x^*) + (x_n - x^*)\varphi'(x^*)$$

або з врахуванням, що $x = \varphi(x)$

$$\varepsilon_{n+1} = \varepsilon_n \varphi'(x^*) \qquad (5.5.2)\ )$$

Очевидно якщо в (5.5.2) $|\varphi'(x)| \le 1$, то $\varepsilon_{n+1} \le \varepsilon_n$, тобто метод збігається і швидкість збіжності є лінійною. Отримане співвідношення (5.5.2) дозволяє сформулювати достатню умову збіжності метода простої ітерації:

якщо $|\varphi'(x)| \le 1$ *для всіх* $x \in [a,b]$, *то ітерації* $x_n = \varphi(x_{n-1})$ *будуть збігатися до єдиної нерухомої точки* $x^* \in [a,b]$ *і точки, які задовольняють цій умові називаються точками тяжіння.*

Таким чином на першому етапі розв'язку початкова задача приводиться до виду ( ), а далі до виконання умови точності $|x_n - x_{n-1}| \le \varepsilon$ проводяться уточнення значення кореня $x_n = \varphi(x_{n-1})$.

В залежності від знака похідної $\varphi'(x)$ існує два виду збіжності до кореня : монотонна при $\varphi'(x) > 0$ (рис.5.1) і коливальна при $\varphi'(x) < 0$ (рис.5.2)

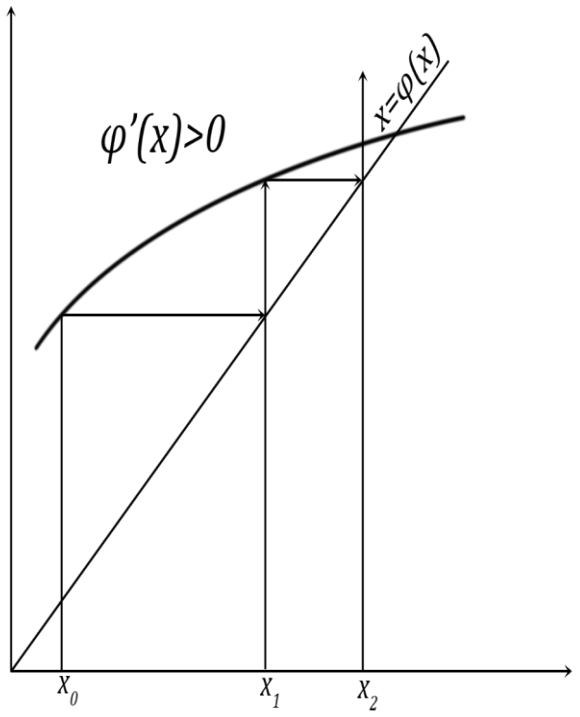


Рис. 5.1 - Випадок монотонної збіжності розв’язку метода простої ітерації

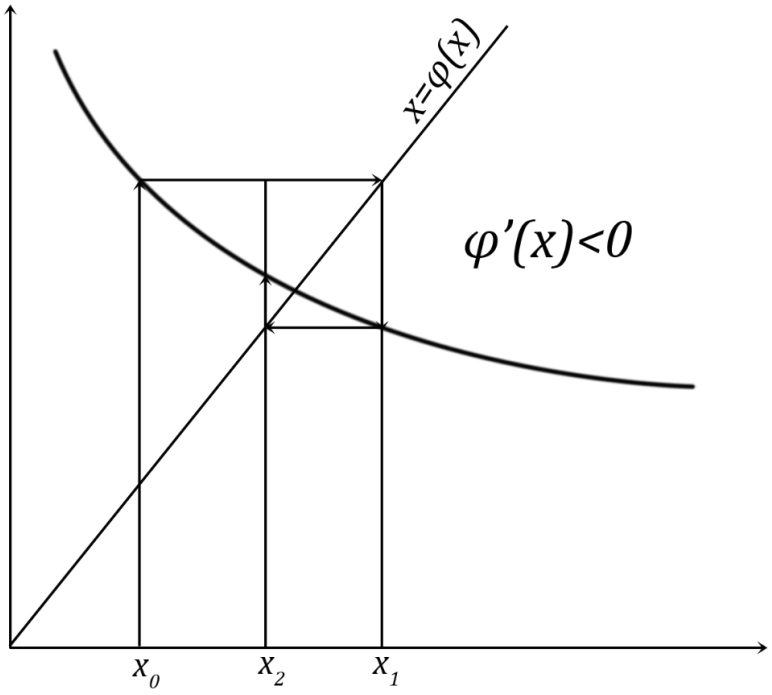


Рис. 5.2 - Випадок коливальної збіжності розв’язку метода простої ітерації

Якщо ж в деякій точці $|\varphi'(x)|>1$, то $\varepsilon_{n+1}>\varepsilon_n$ і ітераційний процес розбігається

. , рис. 5.3 ,то така точка називається точкою відштовхування .

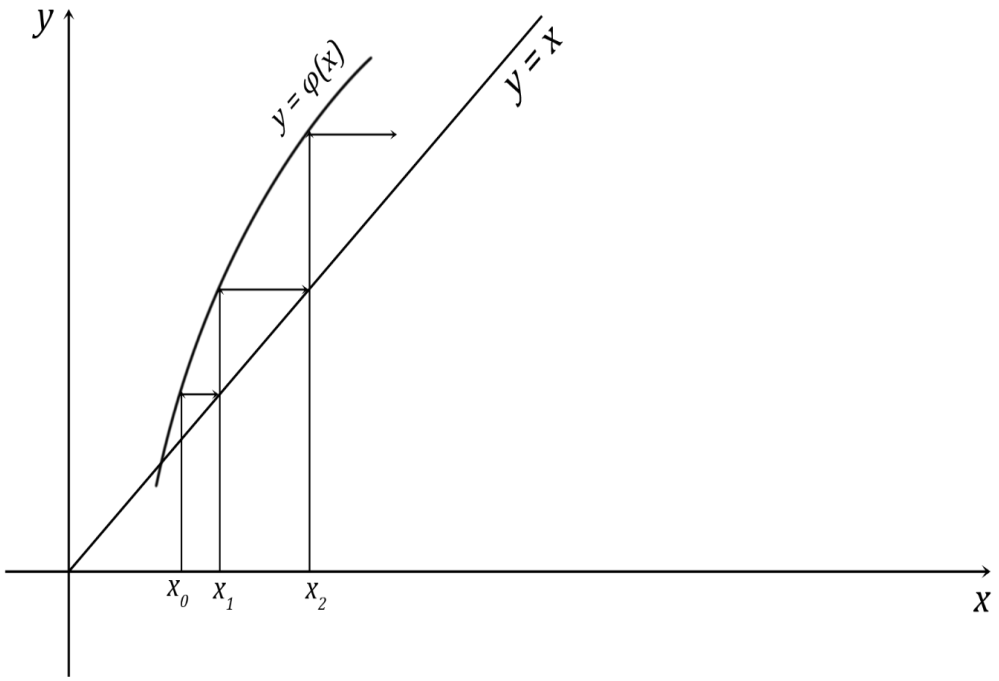


Рис. 5.3 Випадок розбіжності розв’язку метода простої ітерації

Існують спеціальні способи прискорення збіжності. Розглянемо один з них. Нехай процес збіжності наближено представити у вигляді геометричної прогресії

$$x_n - x = Cq^n \qquad (5.5.3)$$

і є три послідовні наближення кореня. Тоді в (5.5.3) можна виключити невідому праву частину

$$\frac{x_n - x}{x_{n-1} - x} = \frac{x_{n+1} - x}{x_n - x}$$

Після розкриття цієї пропорції отримуємо уточнене значення кореня без обчислення самої функції у вигляді наступної формули

$$x^* = \frac{x_{n-1}x_{n+1} - x_n^2}{x_{n-1} + x_{n+1} - 2x_n} \qquad (5.5.4),$$

яка має назву перетворень Ейткена.

Для покращення збіжності цілком природно побудувати обчислювальний процес, щоб кожне нове знайдене покращення відразу вводилося би в обчислення і подальші наближення знаходилися з врахуванням вже зробленого покращення

$$x_{n+1}^{*} = \frac{x_n^{*}\varphi\left[\varphi(x_n^{*})\right] - \varphi^2(x_n^{*})}{\varphi\left[\varphi(x_n^{*})\right] - 2\varphi(x_n^{*}) + x_n^{*}}$$

Описаний метод прискорення збіжності називається процесом Стеффенсона.

Інтервалом невизначеності (a,b) називається інтервал, який містить корінь даного рівняння. Кінці інтервалу знаходяться з умови $f(a)f(b) \le 0$.

Якщо вісь ox у точці-корені не пересікає графік $f(x)$, а є дотичною до нього, то поняття інтервалу невизначеності не має змісту, а знаменник у формулі методу Ньютона обертається в нуль. Ця ситуація виникає у випадку кратних коренів і приведені методи застосувати не можна.

Метод половинного ділення зводиться до зменшення на кожній ітерації довжини інтервалу не визначеності до виконання умови точності $|b_n - a_n| \le \varepsilon,$ де n номер ітерації.

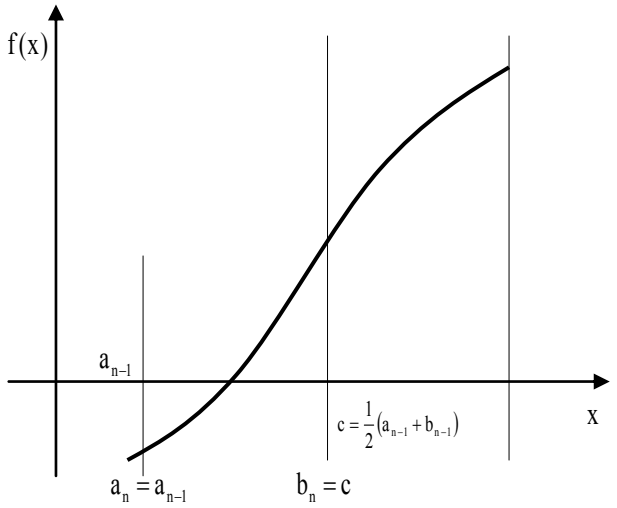


Рис 5.4- Ідея метода половинного ділення

Формування інтервалу невизначеності на ітерації проводиться за формулами

$$f(a_k)\,f(c) \le 0 \Rightarrow a_{k+1} = a_k, b_{k+1} = c.$$
$$f(b_k)\,f(c) \le 0 \Rightarrow a_{k+1} = c, b_{k+1} = b_{k+1}.$$

де $c = \frac{1}{2}(a_k + b_k)$, тобто на кожній ітерації довжина інтервалу зменшується в два рази (рис 5.4) , тобто метод має лінійну швидкість збіжності.

Слабким місцем алгоритму половинного ділення є те, що в ньому враховуються лише знак $f(x)$, а модулі її значень не беруться до уваги. Якщо ж функція $f(x)$ поводить себе «достатньо добре», модулі також мають інформативну цінність, і представляється доцільним використовувати їх при виборі наступного наближення. Орієнтуючись на значення , а не тільки лиш на знаки $f(x)$, можна будувати моделюючі функції, нулі яких обчислюються легко. В двох наступних вихідне нелінійне рівняння наближено замінюється лінійним, розв'язок якого знаходиться безпосередньо. Для знаходження кореня ***методом січних*** як і в методі половинного ділення теж необхідно мати інтервал невизначеності . Якщо його створюють два наближені значення x_{n-1} і x_n, то всередині інтервалу функція заміняється своїм лінійним наближенням

$$f(x) \approx f(x_n) + \frac{f(x_n) - f(x_{n-1})}{x_n - x_{n-1}}(x - x_n),$$

яке є рівнянням січної (рис. 5.5), і визначається корінь цього наближення

$$x^* = x_n - \frac{x_n - x_{n-1}}{f(x_n) - f(x_{n-1})} f(x_n)$$

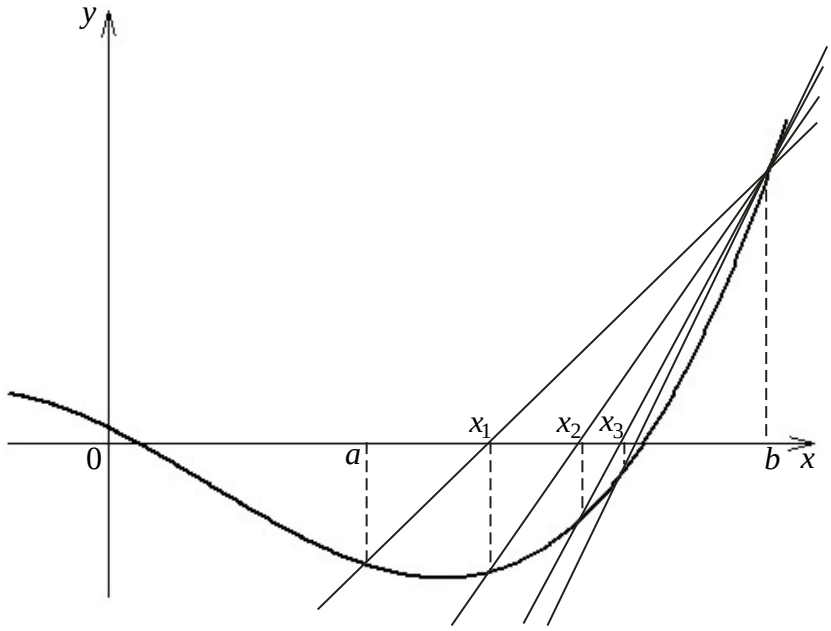


Рис. 5.5.- Метод січних розв'язку нелінійного рівняння

Якщо вести позначення $\Delta_n = \dfrac{x_n - x_{n-1}}{f(x_n) - f(x_{n-1})} f(x_n)$, то робоча формула методу і умова досягнення точності матимуть відповідно форму

$$x^* = x_n - \Delta_n, \quad \|\Delta_n\| \le \varepsilon$$

Після застосування методо є три точки x^*, x_n, x_{n-1} і щоб перейти до наступної ітерації треба залишити дві з них, що утворюють інтервал невизначеності.

Швидкість збіжності метода січних складає $p = 1 + \dfrac{\sqrt{5}-1}{2} \approx 1{,}61803$, що суттєво вище ніж у методі половинного ділення.

В ***методі Ньютона*** (методі дотичних) визначається безпосередньо сам корінь рівняння, а не інтервал до якого він належить (рис. 3.3). Якщо в околі точки x_n функцію $f(x)$ наблизити за допомогою лінійної частини ряду Тейлора

$$f(x_n + h) \approx f(x_n) + hf'(x_n) = 0$$

то $h = -\dfrac{f(x_n)}{f'(x_n)}$, і ітераційний процес будується за формулою

$$x_{n+1} = x_n - \frac{f(x_n)}{f'(x_n)},$$

або як це було зроблено для методу січних $\Delta_n = \dfrac{f(x_n)}{f'(x_n)}$,

маємо

$$x_{n+1} = x_n - \Delta_n, \quad \|\Delta_n\| \le \varepsilon$$

У тому разі коли обчислення похідної зв'язане з труднощами, використовується її скінчено-різницева апроксимація

$$f'(x_k) \approx \frac{f(x_k) - f(x_{k-1})}{x_k - x_{k-1}},$$

після підстановки якої в формула метода Ньютона отримуємо робочу формулу методу січних.

Для збіжності методу Ньютона необхідно виконання наступних умов:

1). Початкове наближення мусить бути достатньо близьким до точного.

2). Друга похідна повинна бути обмеженою $|f''(x)| \le M$.

3). Перша похідна не повинна бути надто малою $|f'(x)| \ge \varepsilon$.

В наслідок порушення першої умови число ітерацій для визначення кореня може бути досить великим. Якщо не виконується друга умова, то лінеаризація функції $f(x)$, може бути грубою, що приводить до тих наслідків, що й у попередньому випадку. Третє обмеження є найбільш суттєвим, постільки використання формули метода стає неможливим. Але досвід для практичної реалізації методу, дозволяє зробити висновок, що для досить надійної його роботи достатньо виконання тільки останньої умови.

В формулу методі Ньютона входить не тільки значення функції, а і значення першої похідної, тобто використовується більше інформації про функцію, ніж у методі половинного ділення і відповідно швидкість збіжності є вище. За допомогою ряду Тейлора, як це робилося при дослідженні збіжності методу простої ітерації, можна показати , що метод Ньютона збігається квадратично.

Метод параболічної апроксимації має в своїй основі наближену заміну функції f квадратичною,тобто початкове рівняння замінюється на квадратне

$$f(z) = az^2 + bz + c = 0, \qquad (5.5.4)$$

яке легко розв'язується в радикалах. Для побудови рівняння у формі(5.5.4) запишемо по трьом останнім наближенням x_n, x_{n-1}, x_{n-2} запишемо інтерполяційний поліном Ньютона

$$f(x) = f_n + (x - x_n) f(x_n, x_{n-1}) + (x - x_n)(x - x_{n-1}) f(x_n, x_{n-1}; x_{n-2})$$

і після введення позначення $z = x - x_n$ коефіцієнти рівняння (5.5.4) дорівнюють

$$a = f(x_n, x_{n-1}; x_{n-2}) \ , b = a(x_n - x_{n-1}) + f(x_n, x_{n-1}) \ \ c = f_n \qquad (5.5.5)$$

Порядок збіжності методу параболічної апроксимації займає проміжне положення між значеннями цієї характеристики методів січних і Ньютона і складає $1{,}839$.

Досі розглядалися методи пошуку тільки якогось одного кореня рівняння. Виникає природне питання як знайти інші корені , якщо їх існує декілька.

Нехай x^* розв'язок рівняння $f(x^*) = 0$. Тоді початкове рівняння можна записати очевидним чином у формі

$$f(x) = (x - x^*)\varphi(x) = 0$$

і пошук наступного кореня зводиться до розв'язку рівняння

$$\varphi(x) = \frac{f(x)}{x - x^*} = 0.$$

Очевидно, що описана процедура може бути повторена необхідне число разів.

5.6 Дійсні алгебраїчні рівняння.

Рівняння виду

$$P_n(x) = a_0 x^n + a_1 x^{n-1} + a_2 x^{n-2} + \ldots + a_{n-1} x^1 + a_n = 0 \qquad (5.6.1)$$

де коефіцієнти a дійсні числа називаються дійсними алгебраїчними рівняннями. Формально (5.6.1) є частковим випадком нелінійного рівняння і його розв'язок можна знайти за допомогою розглянутих методів. Але для розв'язку алгебраїчних рівнянь розроблені спеціальні методи, що обумовлено наступними причинами. По-перше, ці рівняння можуть мати кратні корені, по-друге, корені можуть бути комплексними і розглянуті методи у цих випадках не "працюють". По-третє, будь-який метод, що враховує специфіку задачі, є більш ефективним ніж універсальний.

З вищої алгебри відомо, що коли $\tilde{x}$ корінь рівняння (5.6.1), то

$$P_n(x) = (x - \tilde{x})P_{n-1}(x)\ , \qquad (5.6.2)$$

де $P_{n-1}(x) = b_0 x^{n-1} + b_1 x^{n-1} + \ldots + b_{n-3} x^2 + b_{b-2} x + b_{n-1}$.

Цей результат є основою м*етода виділення лінійного множника.* Нехай x_k, x_{k+1} наближення значення кореня на сусідніх ітераціях. Тоді

$$P_n(x) = (x - x_k)(b_0 x^{n-1} + b_1 x^{n-1} + \ldots + b_{n-3} x^2 + b_{b-2} x) + b_{n-1}(x - x_{k+1}) \quad (5.6.3)$$

Якщо виконується умова точності

$$\left| x_{k+1} - x_k \right| \le \varepsilon\ , \qquad (5.6.4)$$

то множник $(x - x_k)$ можна винести за дужки і отримуємо необхідний результат (5.6.2). В протилежному випадку значення кореня x_{k+1} на наступній ітерації визначається з умов

$$P_n(0) = -b_{n-1}x_{k+1} \quad P_n(x_k) = b_{n-1}(x_k - x_{k+1})$$

Звідси знаходяться $b_{n-1} = \dfrac{P_n(x_k) - P_n(0)}{x_k}$ і

$$x_{k+1} = \frac{P_n(0)}{P_n(0) - P_n(x_k)} x_k$$

Беручи до уваги, що $P_n(0) = a_n$ маємо остаточну формулу метода

$$x_{k+1} = -\frac{a_n}{a_0 x_k^{n-1} + a_1 x_k^{n-1} + \ldots + a_{n-1}}$$

Порівнянням коефіцієнтів при однакових ступенях x, отримуємо ланцюжок формул для послідовного визначення b_i.

$$\begin{aligned} a_0 &= b_0 \\ a_1 &= -x_k b_0 + b_1 \\ a_2 &= -x_k b_1 + b_2 \\ &\ldots\ldots\ldots\ldots \\ a_i &= -x_k b_{i-1} + b_i \end{aligned} \qquad (5.6.5)$$

Після досягнення умови (5.6.4) метод застосовується до рівняння $P_{n-1}(x) = 0$ і т. д..

Метод виділення квадратичного множника (метод Ліна) дозволяє знайти як дійсні так і комплексні корені рівняння (5.6.1). У цьому випадку рівняння приводиться до вигляду

$$P_n(x) = (x^2 + px + q)(b_0 x^{n-2} + b_1 x^{n-3} + \ldots + b_{n-3}x + b_{b-2}) \qquad (5.6.6)$$

де коефіцієнти b_i визначаються тим же шляхом, що й формулі (5.4.5)

$$a_0 = b_0$$

$$a_1 = b_1 + pb_0$$

$$a_2 = b_1 + pb_1 + qb_0$$

. . . . . .

$$a_i = b_i + pb_{i-1} + qb_{i-2}$$

Якщо параметри p, q є такими, що корені рівняння $x^2 + px + q = 0$ не будуть коренями (5.4.1), то співвідношення (5.4.5) повинно містити лінійний залишок і замість (5.5.6) маємо

$$P_n(x) = (x^2 + px + q)(b_0 x^{n-2} + b_1 x^{n-3} + \ldots + b_{n-3}x + b_{b-2}) + \\ + (x + p)b_{n-1} + b_n$$

Поставимо вимогу, щоб додатки в які входять коефіцієнти b_{n-2}, b_{n-1}, b_n , мали структуру

$$b_{n-2}(x^2 + px + q) + (x + p)b_{n-1} + b_n = x^2 + p_{k+1}x + q_{r+1}$$

Порівнюючи відповідні коефіцієнти, отримуємо формули уточнення параметрів p_{k+1}, q_{k+1}

$$p_{k+1} = p_k + \frac{b_{n-1}(p_k, q_k)}{b_{n-2}(p_k, q_k)}$$

$$q_{k+1} = q_k + \frac{b_{n-1}(p_k, q_k) + b_n(p_k, q_k)}{b_{n-2}(p_k, q_k)}$$

Умовою зупинки є як вимога $|p_{k+1} - p_k| \le \varepsilon,\ |q_{k+1} - q_k| \le \varepsilon$, яка еквівалентна умові $|b_{n-1}| \le \varepsilon,\ |b_n| \le \varepsilon$.

Із всіх методів призначених для рішення *систем нелінійних рівнянь*

$$f_i(x_1, x_2, x_3, \ldots, x_n) = 0 \qquad \text{i} = 1,2,\ldots,\text{n} \qquad (5.6.7)$$

найбільш поширеним є метод Ньютона. Цей метод має високу швидкість збіжності і не вимагає від функцій в лівій частині ніяких спеціальних властивостей окрім існування перших похідних. Ідеєю метода є зведення

систем нелінійних рівнянь до послідовності СЛАР за допомогою розкладення в ряд Тейлора функцій у лівій частині з утриманням тільки лінійної складової. Якщо $x^{(k)}$ наближення на k- ій ітерації, то для наступного наближення $x^{(k+1)}) = x^{(k)} + \Delta$ маємо

$$f_i(x_1 + \Delta_1, x_2 + \Delta_2, x_3 + \Delta_3, \ldots, x_n + \Delta_n) = 0 \qquad \text{i} = 1,2,\ldots,\text{n}$$

Після лінеаризації (5.6.7) отримуємо систему відносно компонент вектора $\Delta(\Delta_1, \Delta_2, \Delta_3, \ldots, \Delta_n)$

$$\frac{\partial f_i}{\partial x_1}\Delta_1 + \frac{\partial f_i}{\partial x_2}\Delta_2 + \frac{\partial f_i}{\partial x_3}\Delta_3 + \ldots + \frac{\partial f_i}{\partial x_n}\Delta_n = -f_i(x_1, x_2, x_3, \ldots, x_n)$$

де похідні обчислюються у точці $x^{(k)}(x_1^{(k)}, x_2^{(k)}, \ldots, x_n^{(k)})$.

5.6 ЛАБОРАТОРНА РОБОТА № 5 **Метод виключень Гауса**

Завдання до лабораторної роботи. Для системи лінійних алгебраїчних рівнянь $AX = B$, що задається матрицю A і вектором правої частини B, знайти вектор невідомих X за допомогою метода виключення Гауса і зробити перевірку отриманого результату. Цим же методом обчислити визначник матриці і знайти обернену матрицю.

Методичні вказівки до виконання лабораторної роботи.

Робота виконується у наступній послідовності.

– зчитати матрицю системи і вектор правих частин з попередньо створеного файлу:

```
For i:=nb to ne do
For j:=nb to ne do read(mm,a[i,j]);
For i:=nb to ne do readln(mm,b[i]);
```

– запам’ятати ці величини з новими іменами для проведення перевірки;

– створити допоміжну одиничну матрицю D для обчислення оберненої матриці A^{-1} ;

– за формулами

$$a_{ij} = a_{ij} - \frac{a_{kj}}{a_{kk}} a_{ik} \text{ і } b_i = b_i - \frac{b_k}{a_{kk}} a_{ki}$$

провести пряму ходу методу Гауса як потрійний цикл, у якому на зовнішньому рівні k – поточний номер змінної, що виключається, і – номер поточного рядка, j – номер поточного стовпця. Параметр k міняється від 1 до n-1, а параметри і та j від k до n.

– всередині циклу по змінним k та і разам з обробкою компонент вектора B по аналогічній формулі перерахувати і-ту компоненту матриці D у всіх стовпцях.

Технічно це досягається додаванням циклу

for m:=1 to n do d[m,i]:=d[m,i]-s*d[m,k];

Після закінчення прямої ходи матриця буде приведена до трикутного виду і можна обчислити її визначник згідно формулі

$$\det(A) = \prod_{k=1}^{n} a_{kk}$$

При реалізації формул оберненої ходи параметр циклу змінюється від n до одиниці і тому використовується оператор циклу

for i:=n downto 1 do

Щоб уникнути помилок на оберненій ході обчислювати компоненти вектора X по формулі

$$x_k = \frac{1}{a_{kk}} \left(b_k - \sum_{j=k+1}^{n} a_{kj} x_j \right) \qquad (3.8)$$

і елементи оберненої матриці краще в окремих циклах. Цикл Технічно це означає, що для визначення A^{-1} треба зробити ще один (самий зовнішній), в якому параметр m буде змінюватися від 1 до n. І відповідна сума в ( ) буде обчислюватися як

```
for k:=i+1 to n do s:=s+a[i,k]*au[m,k];
```

Контрольні питання.

В чому полягає ідея виключень Гауса?

Що називається провідним елементом?

Чому для перевірки розв’язку СЛАР методом Гауса необхідно зберігати вихідні значення матриці і вектора правої частини?

Чому обчислення оберненої матриці за методом Гауса має перевагу над її обчисленням безпосередньо по формулам лінійної алгебри?

Коли при розв’язку СЛАР стає відомо, що матриця системи є виродженою?

Чому метод прогонки потребує меншого обсягу обчислень, ніж розв’язок цієї ж СЛАР методом Гауса?

Варіанти завдань до лабораторної роботи.

1. $\begin{pmatrix} 1 & 2 & 3 & 7 & -2 & -1 \\ 5 & 7 & 2 & 3 & 1 & 5 \\ 3 & 1 & 4 & 0 & 2 & 0 \\ 8 & 11 & 4 & -3 & 1 & 9 \\ 13 & 22 & 1 & -1 & 2 & 3 \\ 7 & -5 & 2 & 3 & 8 & 9 \end{pmatrix} \begin{pmatrix} 2 \\ 3 \\ 4 \\ 1 \\ 1 \\ 2 \end{pmatrix}$

2. $\begin{pmatrix} 1 & -2 & 3 & 7 & -2 & -1 \\ 5 & 0 & 2 & 0 & 1 & 5 \\ 3 & 1 & 4 & 0 & 2 & 0 \\ 8 & -1 & 4 & -3 & 1 & 9 \\ -3 & 12 & 1 & -1 & 2 & 3 \\ 7 & -5 & 2 & 4 & 8 & 2 \end{pmatrix} \begin{pmatrix} 2 \\ 6 \\ 4 \\ 4 \\ 2 \\ 1 \end{pmatrix}$

3. $\begin{pmatrix} 1 & -2 & 3 & 7 & -2 & -1 \\ 9 & 1 & 2 & 0 & 1 & 5 \\ 3 & 8 & 4 & 0 & 2 & 0 \\ 4 & 11 & 4 & 13 & 6 & -7 \\ -3 & 8 & 1 & -1 & 2 & 3 \\ 7 & -5 & 2 & 4 & 8 & 2 \end{pmatrix} \begin{pmatrix} 1 \\ 2 \\ 3 \\ 3 \\ 2 \\ 0 \end{pmatrix}$

4. $\begin{pmatrix} 4 & -2 & 5 & 6 & -2 & -1 \\ 0 & 1 & 2 & 0 & 1 & 5 \\ 3 & 3 & 0 & 0 & 2 & 0 \\ 0 & 21 & 4 & 9 & 3 & -3 \\ -2 & 8 & 1 & -1 & 2 & 3 \\ 0 & -5 & 2 & 4 & 8 & 12 \end{pmatrix} \begin{pmatrix} 7 \\ -9 \\ 0 \\ -1 \\ 1 \\ -6 \end{pmatrix}$

5. $\begin{pmatrix} 4 & -3 & 2 & -1 & 0 & -1 \\ 7 & 8 & 0 & 4 & 11 & 5 \\ 3 & 3 & 1 & 3 & 2 & 7 \\ 1 & -2 & 0 & 9 & 3 & -3 \\ -2 & 8 & 1 & -1 & 2 & 3 \\ 0 & -5 & 0 & 4 & 8 & -2 \end{pmatrix} \begin{pmatrix} 1 \\ 3 \\ 5 \\ 7 \\ 7 \\ 1 \end{pmatrix}$

6. $\begin{pmatrix} -1 & 0 & 1 & 1 & 1 & 0 \\ 0 & 1 & 1 & 1 & 0 & 1 \\ 0 & 1 & 0 & 3 & 1 & 1 \\ 1 & 0 & 1 & 1 & 0 & 0 \\ 1 & 1 & 1 & 0 & 1 & 1 \\ 0 & 1 & 1 & 1 & 1 & 0 \end{pmatrix} \begin{pmatrix} 3 \\ 3 \\ 2 \\ 2 \\ 1 \\ 1 \end{pmatrix}$

7. $\begin{pmatrix} 4 & -3 & 2 & -1 & 0 & -1 \\ 7 & 0 & 0 & 4 & 1 & -2 \\ 9 & 4 & 0 & 3 & 1 & 7 \\ 1 & 2 & 0 & 0 & 3 & 3 \\ -2 & 5 & 1 & -1 & 0 & -3 \\ 0 & 11 & 0 & 4 & 8 & 0 \end{pmatrix} \begin{pmatrix} 0 \\ 2 \\ 3 \\ 5 \\ 9 \\ 1 \end{pmatrix}$

8. $\begin{pmatrix} 7 & -5 & -2 & 2 & 2 & -7 \\ 2 & 1 & 2 & 4 & 1 & 11 \\ 4 & 4 & 4 & 3 & 4 & -6 \\ 1 & 2 & 9 & 6 & 0 & 3 \\ 0 & 5 & 1 & 33 & 2 & 3 \\ 6 & 3 & 2 & 0 & 8 & 9 \end{pmatrix} \begin{pmatrix} 0 \\ 0 \\ 1 \\ 1 \\ 1 \\ 9 \end{pmatrix}$

9. $\begin{pmatrix} 2 & -5 & -2 & 2 & 2 & -7 \\ 2 & 1 & 2 & 4 & 1 & 77 \\ 6 & 4 & 4 & 3 & 4 & 5 \\ 1 & 2 & 9 & 6 & 0 & 2 \\ 3 & 5 & 1 & 33 & 2 & 8 \\ 0 & 3 & 2 & 0 & 8 & 9 \end{pmatrix} \begin{pmatrix} 2 \\ 5 \\ 3 \\ 2 \\ 7 \\ 1 \end{pmatrix}$

10. $\begin{pmatrix} 5 & -5 & 5 & -5 & 5 & -5 \\ 5 & 1 & 2 & 4 & 11 & 3 \\ 7 & 0 & 4 & 3 & 0 & 5 \\ 2 & 2 & 9 & 6 & 0 & 2 \\ 4 & 5 & 1 & 33 & 2 & 8 \\ -8 & 8 & -8 & 8 & -8 & 8 \end{pmatrix} \begin{pmatrix} 22 \\ 0 \\ 0 \\ 0 \\ 0 \\ 22 \end{pmatrix}$

11. $\begin{pmatrix} 2 & 1 & 2 & 1 & 2 & 1 \\ 5 & 1 & 2 & 4 & 11 & 3 \\ 7 & 0 & 4 & 3 & 3 & 5 \\ 2 & 2 & 9 & 6 & 0 & 2 \\ 5 & 4 & 3 & 0 & 2 & 8 \\ -6 & 6 & -6 & 6 & -6 & 6 \end{pmatrix} \begin{pmatrix} 0 \\ 1 \\ 7 \\ 1 \\ 7 \\ 0 \end{pmatrix}$

12. $\begin{pmatrix} 9 & 4 & 2 & 4 & 2 & 4 \\ 8 & 1 & 2 & 4 & 11 & 3 \\ 7 & 9 & 4 & 3 & 3 & 5 \\ 6 & 2 & 9 & 6 & 0 & 2 \\ 5 & 4 & 3 & 0 & 2 & 8 \\ 4 & 3 & 2 & 1 & -1 & 6 \end{pmatrix} \begin{pmatrix} 2 \\ 3 \\ 8 \\ 8 \\ 3 \\ 2 \end{pmatrix}$

13. $\begin{pmatrix} 1 & 4 & 2 & 4 & 2 & 4 \\ 1 & 1 & 2 & 4 & 11 & 3 \\ 3 & 9 & 4 & 3 & 3 & 2 \\ 4 & 2 & 9 & 6 & 0 & 0 \\ 8 & 4 & 3 & 0 & 2 & 1 \\ 8 & 3 & 2 & 1 & -1 & 0 \end{pmatrix} \begin{pmatrix} 1 \\ 2 \\ 3 \\ 4 \\ 7 \\ 1 \end{pmatrix}$

14. $\begin{pmatrix} 1 & 4 & 2 & 4 & 2 & 4 \\ 1 & 2 & 3 & 4 & 5 & 9 \\ 3 & 9 & 4 & 3 & 3 & 2 \\ 4 & 0 & 9 & 6 & 0 & 0 \\ 8 & 4 & 3 & 0 & 2 & 1 \\ 8 & 3 & 2 & 1 & 1 & 0 \end{pmatrix} \begin{pmatrix} 6 \\ 7 \\ 7 \\ 5 \\ 1 \\ 3 \end{pmatrix}$

15. $\begin{pmatrix} -1 & 4 & 2 & 4 & 2 & 4 \\ 10 & 2 & 3 & 4 & 5 & 9 \\ 33 & 9 & 44 & 3 & 3 & 2 \\ 4 & 0 & 9 & 6 & 0 & 0 \\ 8 & 4 & 3 & 0 & 2 & 1 \\ 18 & 3 & 2 & 1 & 1 & 0 \end{pmatrix} \begin{pmatrix} 8 \\ 1 \\ 2 \\ 6 \\ 2 \\ 6 \end{pmatrix}$

16. $\begin{pmatrix} -1 & 0 & 2 & 1 & 2 & 0 \\ 0 & 2 & 3 & 1 & 0 & 1 \\ 0 & 9 & 0 & 3 & 3 & 2 \\ 1 & 0 & 1 & 1 & 0 & 0 \\ 1 & 4 & 3 & 0 & 2 & 1 \\ 0 & 3 & 2 & 1 & 1 & 0 \end{pmatrix} \begin{pmatrix} 2 \\ 3 \\ 7 \\ 1 \\ 2 \\ 2 \end{pmatrix}$

17. $\begin{pmatrix} 1 & 4 & 2 & 4 & 2 & 14 \\ 2 & 1 & 2 & 4 & 11 & 10 \\ 3 & 9 & 4 & 3 & 3 & 9 \\ 4 & 2 & 9 & 6 & 0 & 8 \\ 5 & 4 & 3 & 0 & 2 & 7 \\ 6 & 3 & 2 & 1 & -1 & 6 \end{pmatrix} \begin{pmatrix} 8 \\ 3 \\ 4 \\ 3 \\ 1 \\ 8 \end{pmatrix}$

18. $\begin{pmatrix} 2 & 4 & 2 & 4 & 2 & 4 \\ 5 & 1 & 2 & 4 & 11 & 3 \\ 7 & 9 & 4 & 3 & 3 & 5 \\ 2 & 2 & 9 & 6 & 0 & 2 \\ 5 & 4 & 3 & 0 & 2 & 8 \\ 7 & 6 & -7 & 6 & -7 & 6 \end{pmatrix} \begin{pmatrix} 7 \\ 7 \\ 4 \\ 3 \\ 1 \\ 5 \end{pmatrix}$

19. $\begin{pmatrix} 3 & -5 & -2 & 2 & 2 & 6 \\ 5 & 1 & 2 & 4 & 1 & 13 \\ 7 & 0 & 4 & 3 & 0 & 5 \\ 2 & 2 & 9 & 6 & 0 & 2 \\ 4 & 5 & 1 & 33 & 2 & 8 \\ 8 & 3 & 2 & 1 & 8 & 9 \end{pmatrix} \begin{pmatrix} 4 \\ 4 \\ 4 \\ 0 \\ 0 \\ 1 \end{pmatrix}$

20. $\begin{pmatrix} 9 & -3 & 2 & -1 & 3 & -1 \\ 7 & 1 & 2 & 4 & 1 & -2 \\ 9 & 4 & 4 & 3 & 1 & -6 \\ 1 & 2 & 9 & 6 & 3 & 3 \\ -2 & 5 & 11 & 8 & 0 & 3 \\ 10 & 11 & 0 & 4 & 8 & 9 \end{pmatrix} \begin{pmatrix} 2 \\ 3 \\ 0 \\ 2 \\ 9 \\ 2 \end{pmatrix}$

21. $\begin{pmatrix} 4 & -3 & 2 & -1 & 0 & -1 \\ 7 & 1 & 0 & 4 & 1 & 0 \\ 3 & 3 & 0 & 3 & 2 & 7 \\ 1 & -2 & 0 & 0 & 3 & 3 \\ -2 & 5 & 1 & -1 & 0 & -3 \\ 0 & -5 & 0 & 4 & 8 & -2 \end{pmatrix} \begin{pmatrix} 1 \\ 3 \\ 4 \\ 7 \\ 1 \\ 2 \end{pmatrix}$

22. $\begin{pmatrix} 4 & -2 & 5 & 6 & -2 & -1 \\ 7 & 1 & 2 & 4 & 1 & 5 \\ 3 & 3 & 1 & 3 & 2 & 7 \\ 1 & 21 & 4 & 9 & 3 & -3 \\ -2 & 8 & 1 & -1 & 2 & 3 \\ 10 & -5 & 2 & 4 & 8 & 12 \end{pmatrix} \begin{pmatrix} 6 \\ 8 \\ 4 \\ 2 \\ 4 \\ 0 \end{pmatrix}$

23. $\begin{pmatrix} 9 & -2 & 4 & 7 & -2 & -1 \\ 9 & 1 & 2 & 0 & 1 & 5 \\ 0 & 6 & 0 & 0 & 2 & 0 \\ 0 & 11 & 4 & 9 & 3 & -7 \\ -3 & 8 & 1 & -1 & 2 & 3 \\ 7 & -5 & 2 & 4 & 8 & 32 \end{pmatrix} \begin{pmatrix} 0 \\ 7 \\ 0 \\ 1 \\ 1 \\ 0 \end{pmatrix}$

24. $\begin{pmatrix} 1 & -2 & 3 & 7 & -2 & -1 \\ 0 & 1 & 2 & 0 & 1 & 5 \\ 3 & 1 & 4 & 0 & 2 & 0 \\ 4 & -1 & 4 & -3 & 6 & 9 \\ -3 & 8 & 1 & -1 & 2 & 3 \\ 7 & -5 & 2 & 4 & 8 & 2 \end{pmatrix} \begin{pmatrix} 0 \\ 4 \\ 7 \\ 1 \\ 0 \\ 9 \end{pmatrix}$

25. $\begin{pmatrix} -4 & 2 & 3 & 7 & -2 & -1 \\ 0 & 7 & 2 & 3 & 1 & 5 \\ 1 & 1 & 4 & 0 & 2 & 0 \\ 8 & 13 & 4 & -3 & 1 & 9 \\ 13 & 22 & 1- & 1 & 2 & 3 \\ 7 & -5 & 2 & 3 & 8 & 17 \end{pmatrix} \begin{pmatrix} 0 \\ 1 \\ 1 \\ 0 \\ 0 \\ 9 \end{pmatrix}$

5.7 ЛАБОРАТОРНА РОБОТА № 6 **Визначення власних чисел і векторів матриці**

Завдання до лабораторної роботи. Знайти найбільше власне число матриці і відповідний йому власний векторів використавши ітераційний метод. Матриця A береться за варіантом лабораторної роботи № 5.

Методичні вказівки до виконання лабораторної роботи.

Робота виконується у наступній послідовності.

1. зробити вихідну матрицю симетрично, при цьому її елементи, що розташовані вище головної діагоналі не змінюються.

2. В якості початкового наближення $x_i^{(0)}$ взяти довільний вектор, але не можна брати нульовий вектор, бо отримаємо безкінечний обчислювальний процес.

3. В циклі типу Repeat – Until до виконання умови точності виконується

1) обчислюється допоміжний вектор

$$y_m = \sum_{k=1}^{n} a_{mk} x_k^{(r)},$$

де r – номер попередньої ітерації;

2) знаходиться номер M, найбільшої по абсолютній величині компоненти y, яка береться в якості наступного наближення для власного числа $\lambda^{(r+1)} = y_M$;

3) в циклі, в якому параметр змінюється від 1 до n одночасно обчислюється величина відхилення по нормі поточного наближення від попереднього і визначається наступне наближення. Для цього вводиться допоміжна змінна $t = \frac{y_m}{\lambda^{(r+1)}}$, обчислюється відхилення для компоненти як $(t - x_m^{(r)})^2$ і підсумовується по всіх компонентах $s = s + (t - x_m^{(r)})^2$. Знаходиться нове наближення $x_m^{(r+1)} = t$.

4. Зовнішній цикл зупиняється при виконані умови $\sqrt{s} \le \varepsilon$.

Контрольні питання.

Які матриці мають виключно дійсні власні числа?

Чи можна матричну проблему розв'язати шляхом виключень Гауса в тих випадках, коли матриця набуває трикутної форми і матрична проблема вирішиться безпосередньо?

Яким методом можна знайти комплексні власні числа?

5.8 ЛАБОРАТОРНА РОБОТА № 7 **Розв'язок нелінійних рівнянь**

Завдання до лабораторної роботи. Знайти корінь нелінійного рівняння у вказаному інтервалі методом половинного ділення, методом січних і методом Ньютона. Перевірити правильність отриманих рішень і порівняти швидкість збіжності цих методів.

Методичні вказівки до виконання лабораторної роботи.

Приклад. Розглянемо роботу методу половинного ділення для рівняння з правою частиною

$$f(x) = e^{x+2} - x^2.$$

Точки a=-2 і b=3 будуть кінцями інтервал невизначеності, оскільки f(a)=-3.00, f(b)=139.413.

На першому кроці маємо c=0.5 і f(c)=2.9278. Тобто новий інтервал невизначеності буде визначатися тепер точками a і c, в яких функція має різний

знак. В наступних двох ітераціях ліва точка лишається незмінною і відкидається права половина інтервалу

a=-2.000 b=0.5000 c=-0.7500 f(a)=-3.0000 f(b)=11.9325 f(c)=2.9278

a=-2.000 b=-0.7500 c=-1.3750 f(a)=-3.0000 f(b)= 2.9278 f(c)=-0.0224

На четвертій і п’ятій ітераціях маємо обернену картину

a=-1.375 b=-0.7500 c=-1.0625 f(a)=-0.0224 f(b)= 2.9278 f(c)= 1.4247

a=-1.375 b=-1.0625 c=-1.2188 f(a)=-0.0224 f(b)= 1.4247 f(c) 0.6988

Всього ж для досягнення точності $\varepsilon = 10^{-3}$ у цій задачі треба виконати 13 ітерацій.

При розв’язку цієї ж задачі методом Ньютона, коли в якості початкового наближення вибирати по черзі ліву, праву і середню точки початкового інтервал знадобляться відповідно 2,6 і 3 ітерації. Що демонструє наскільки швидше буде отримано рішення при вдалому початковому наближенні. У нашій задачі точний розв’язок -1.370154 значно ближче розташований до лівого кінця.

Контрольні питання.

Чим вимірюється швидкість збіжності метода?

Як перевірити правильність знайденого розв’язку?

Який з методів: половинного ділення, січних чи Ньютона швидше збігається і чому?

Чи можна для одного ж того ж рівняння отримати два або більше правильних рішень?

Яким умовам повинні відповідати кінцеві точки інтервалу невизначеності?

Варіанти завдань до лабораторної роботи

Варіант	Функція	Інтервал
1	$f(x) = 0.5x^3 - \sin x$	[-2; 3]
2	$f(x) = \sin x - 0.5x + 1$	[-5; 5]
3	$f(x) = x^2 - \sin(x + 0{,}2)$	[-5; 5]
4	$f(x) = x^2 - 5x + 2\sin x + 1$	[-5;5]

5	$f(x)=\sin x(x-\sin x)$	[-5; 5]
6	$f(x)=x\cos x+1$	[-5; 5]
7	$f(x)=x^4-\cos x$	[-2; 0]
8	$f(x)=\cos(\sin x)-0{,}5x$	[-2; 2]
9	$f(x)=e^x-0.5*x^2$	[-2; 2]
10	$f(x)=x-\ln(x+10)$	[0; 5]
11	$f(x)=4x^2-e^x$	[0; 3]
12	$f(x)=3x^2-e^{(x-1)}$	[0; 3]
13	$f(x)=3x^3-\ln(x^2+2)$	[-3; 3]
14	$f(x)=\sin x-\ln(x^2+2)+x$	[-3; 3]
15	$f(x)=0.5x^3-e^x\sin x+1$	[-3; 3]
16	$f(x)=\frac{1}{2}x^3-e^x\sin(x+1)$	[-2; 3]
17	$f(x)=x^2+e^x\cos(x+1)$	[-3; 3]
18	$f(x)=x^2+(1-x)e^x$	[-3; 3]
19	$f(x)=x\cos s(x+1)+\ln(x^2+1)-1$	[-3; 3]
20	$f(x)=x\sin(x+1)+\ln(x^2+1)-1$	[-3; 3]
21	$f(x)=x^2+(1-x^2)e^x$	[-3; 3]
22	$f(x)=x^3-2x+(x-1)\sin(x+1)$	[-3; 3]
23	$f(x)=x^3-2x+(x+1)\sin(x-1)$	[-3; 3]
24	$f(x)=x^3-2x+(x+1)\cos(x-1)$	[-3; 3]
25	$f(x)=x^3-x+(x+1)\cos(x+1)$	[-3; 3]

РОЗДІЛ 6 ЗВИЧАЙНІ ДИФЕРЕНЦІАЛЬНІ РІВНЯННЯ

6.1 Постановка задачі

Серед задач, з якими доводиться мати справу в обчислювальній практиці , значну частину складають різноманітні задачі для звичайних диференціальних рівнянь. Такі задачі виникають як безпосередньо при математичному моделюванні багатьох реальних явищ,так і в якості проміжних при розв'язку ряду більш складних математичних задач. При цьому, як правило точний розв'язок задачі, що розглядається, не вдається виразити через елементарні функції. Частка задач, що мають розв'язок у явному вигляді у випадку звичайних диференціальних рівнянь мізерно мала. Звичайно доводиться вдаватися до допомоги наближених методів розв'язку подібних задач.

Диференціального рівняння першого порядку можна записати у вигляді

$$y' = f(t, y)$$

Як відомо це рівняння має сімейство рішень $y(t)$. Наприклад, якщо $f(t, y) = t$, то для довільної константи C функція $y(t) = Ce^t$ є розв'язком. Вибір початкового значення, наприклад $y(0)$, дозволяє знайти цю константу, і тим самим з сімейства рішень визначити одне. Початкове значення залежної змінної може бути заданим для будь якого значення t_0 незалежної змінної. Але часто вважають , що виконано перетворення , яке забезпечує, що $t_0 = 0$. Це не впливає ні на розв'язок ні на методи, застосовані для наближення розв'язку.

В багатьох випадках є більш ніж одна залежна зміна, тоді задача полягає в розв'язку системи рівнянь першого порядку. Наприклад

$$y' = f(t, y, z)$$

$$z' = g(t, y, z)$$

Допустимо, що похідні $\frac{df}{dy}$, $\frac{df}{dz}, \frac{dg}{dy}, \frac{dg}{dz}$ існують на всьому інтервалі інтегрування. розв'язок цієї системи містить дві сталі інтегрування, і, отже ,

необхідно дві додаткові умови, щоб визначити ці константи. Якщо значення y і z вказані при одному й тому ж значенні незалежної змінної t_0, то система буде мати єдине рішення. Задача знаходження значень y і z для значень $t > t_0$ називається *задачею Коші*.

Будь яке звичайне диференціальне рівняння прядку k, яке може бути записане так, що його ліва частина є похідна найвищого порядку, а правій частині ця похідна не фігурує, може бути записане у формі системи з k рівнянь першого порядку шляхом введення $k-1$ нових змінних. Наприклад рівняння $y'' = f(t, y, y')$ може бути записано у вигляді системи

$$z' = f(t, y, z)$$
$$y' = z,$$

де $z'(t) = y''(t)$. В векторних позначеннях це набере наступний вигляд

$$u' = F(t, u),$$

де $u = \begin{pmatrix} z \\ y \end{pmatrix} = \begin{pmatrix} u_1 \\ u_2 \end{pmatrix}$, $F(t,u) = \begin{pmatrix} f \\ z \end{pmatrix} = \begin{pmatrix} f \\ u_1 \end{pmatrix}$

Тобто задачі Коші для систем і рівнянь порядку вище першого приводиться до задачі Коші для Диференціального рівняння першого порядку, а саме

$$y' = f(t, y);\ y(t_0) = y_0 \qquad (6.1)$$

Якщо необхідно знайти розв’язок диференціального рівняння при цьому додаткові задаються при двох або більше значеннях незалежної змінної, то задача називається *крайовою*.

6. 2 Чисельний розв’язок задачі Коші.

Чисельне рішення цієї задачі полягає в обчисленні значень функції в дискретній послідовності точок $y(x_1), y(x_2),...,x(y_n)$. Хоча для чисельного розв’язку звичайних диференціальних рівнянь існує досить багато різних покрокових методів,кожен з них неминуче попадає в одну з чотирьох загальних категорій: методи рядів Тейлора, методи Рунге-Кутти і багато крокові методи

Методи рядів Тейлора. Якщо $y(t)$ - гладкий розв’язок, маємо розкладення шуканої функції по Тейлору

$$y(t+h) = y(t) + hy'(t) + \frac{1}{2!}h^2 y''(t) + ... + \frac{1}{k!}h^k y^{(k)}(t) + ... \qquad (6.2)$$

Метод Ейлера можна розглядати як наближення, що використовує тільки лінійну по h частину (6.3). Поклавши в (6.2) $t = x_n$ з врахуванням (6.1) , отримуємо робочу формулу метода

$$y_{n+1} = y_n + hf(x_n, y_n) \qquad (6.3)$$

З геометричної інтерпретації методу Ейлера (рис 6.1) видно, що точне рішення буде одержано тільки у випадку лінійної функції $f(x, y)$

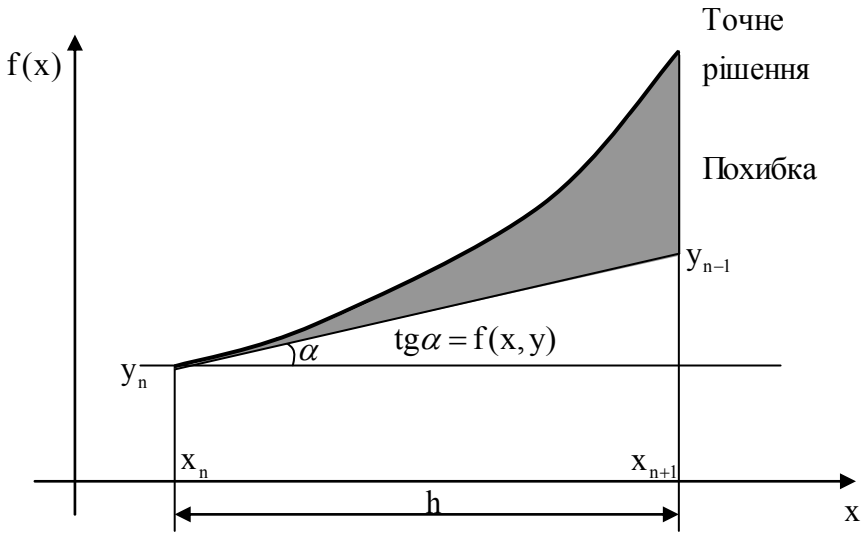


Рис. 6.1.- Крок розв'язку методу Ейлера

Точність методу Ейлера можна суттєво підвищити шляхом покращення апроксимації похідної. Це можна зробити використавши її середнє значення на початку і в кінці під інтервалу. В модифікованому методі Ейлера спочатку обчислюється значення функції в наступній точці по методу Ейлера

$$y^*_{n+1} = y_n + hf(x_n, y_n),$$

яке застосовується для обчислення наближеного значення в кінці підінтервалу $f(x_{n+1}, y^*_{n+1})$. Після обчислення середнього між цим значенням похідної і на початку під інтервалу знаходимо

$$y^*_{n+1} = y_n + \frac{h}{2}\left[f(x_n, y_n) + f(x_{n+1,} y^*_{n+1})\right] \qquad (6.4)$$

Принцип, на якому побудовано модифікований метод Ейлера можна пояснити наступним чином. Якщо наблизити другу похідну скінченої різницею

$$y''(x_n) = \frac{y'(x_n + h) - y'(x_n)}{h}$$

і підставити її в квадратичний відрізок ряду Тейлора, то приходимо до формули (6.4). Тобто цей метод є методом другого порядку.

Якщо можуть бути обчислені похідні функції y більш високого порядку, то можна отримати метод порядку з p, положивши

$$y(x_{n+1}) = y(x_n + h) \approx y(x_n) + hy'(x_n) + \frac{1}{2!}h^2 y''(x_n) + \ldots + \frac{1}{p!}h^p y^{(p)}(x_n) + \ldots (6.5)$$

Перший відкинутий член в (6.5) дає оцінку локальної похибки дискретизації і тому може бути використаним для вибору величині кроку. Похідні $y'(x), y''(x)$ і т.д. можна виразити через часткові похідні функції f. При цьому важливо відрізняти функцію $f(y, x)$ двох незалежних змінних і функцію $f(y(x), x)$ однієї незалежної змінної, отриману підстановкою розв'язку y в f.

В силу співвідношення (6.1) формули для похідних в (6.5) набудуть вигляду

$$y' = f \ ;$$

$$y'' = \frac{\partial f}{\partial x} + \frac{\partial f}{\partial y}\frac{\partial y}{\partial x} = f_x + f_y f$$

$$y''' = f_{xx} + 2 f_{xy} f + f_{yy} f^2 + (f_x + f_y f) f_y$$

де всі обчислення проводяться в точці (x_n, y_n). Слід відмітити, що навіть для поліному f від двох змінних вирази для повних похідних ускладнюються по мірі зростання ступеня. Можна скласти програму , яка обчислює повні похідні для деяких класів функцій. Для диференціальних рівнянь , що визначаються такими функціями , метод рядів може бути досить ефективним, Але в загальному випадку використання таких методів є досить обмеженим

Методи Рунге-Кутти. Ці методи призначені для отримання розв'язку на основі рядів Тейлора, але без явного обчислення похідних, за винятком першої. Шляхом безпосереднього інтегрування $\int_{x_n}^{x_n+h} y'dx = \int_{x_{kn}}^{x_n+h} f(x,y)dx$ з врахуванням умови $y(x_n) = y_n$ розв'язок задачі Коші у точці x_{n+1} можна записати у вигляді

$$y_{n+1} = y_n + \int_{x_n}^{x_n+h} f(x,y)dx \text{ або } y_{n+1} = y_n + \Delta_n \qquad (6.6)$$

Тоді загальна формула чисельних методів рішення задачі Коші може бути представлена у вигляді

$$y_{n+1} = y_n + \Delta_n \qquad (6.7)$$

У методах Рунге-Кутти величина Δ_n обчислюється як

$$\Delta_n \approx \sum_{k=0}^{q} A_k \varphi_k, \quad (6.8)$$

де A_k набір констант, φ_k - набір допоміжних функцій: Якщо ввести ще два набори констант

$$\begin{aligned}
&\alpha_1, \alpha_2, \ldots, \alpha_q \\
&\beta_{10} \\
&\beta_{20}, \beta_{21} \\
&\ldots\ldots\ldots\ldots\ldots\ldots \\
&\beta_{q0}, \beta_{q1}, \ldots, \beta_{qq-1}.
\end{aligned}$$

і визначити φ_q як

$$\begin{aligned}
&\varphi_0 = h \cdot f(x, y) \\
&\varphi_1 = h \cdot f(x + \alpha_1 \cdot h, y + \beta_{10} \cdot \varphi_0) \\
&\varphi_2 = h \cdot f(x + \alpha_2 \cdot h, y + \beta_{20} \cdot \varphi_0 + \beta_{21} \cdot \varphi_1) \\
&\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots \\
&\varphi_q = h \cdot f(x + \alpha_q \cdot h, y + \beta_{q0} \cdot \varphi_0 + \beta_{q1} \cdot \varphi_1 + \ldots + \beta_{qq-1} \cdot \varphi_{q-1})
\end{aligned}$$

то φ_q можна послідовно обчислити по відомих α і β . Постільки (6.5) і (6.7) мають однакову структуру, то параметри A, α, β можна знайти шляхом порівняння коефіцієнтів при однакових ступенях h в (6.5) і в розкладенні Δy в ряд Тейлора в околі $h = 0$. Порядок метода буде визначатися найвищим ступенем h утриманим в цих розкладеннях.

При $q = 0$ необхідно обчислити

$$\begin{aligned}
&\Delta(h) = A_0 h f(x_n, y_n) \qquad \Delta'(h) = A_0 f(x_n, y_n) \\
&\Delta(0) = 0 \qquad\qquad\qquad\quad \Delta'(0) = A_0 f(x_n, y_n).
\end{aligned}$$

Тоді $y_n + hf = y_n + hA_0 f$, звідки визначаємо $A_0 = f(x_n, y_n)$, тобто метод Рунге-Кутти першого порядку і метод Ейлера співпадають.

Аналогічні обчислення при $q = 1$ дають

$$\Delta(h) = A_0 hf(x_n, y_n) + A_1 hf(x_n + \alpha_1 h, y_n + \beta_{10} hf) = A_0 hf(x_n, y_n) + A_1 hf(h)$$

$$\Delta'(h) = A_0 f + A_1 f(h) + A_1 h\{\alpha_1 f_x(h) + \beta_{10} f_y(h) f(h)\} \quad \Phi = \alpha_1 f_x(h) + \beta_{10} f_y(h) f(h)$$

$$\Delta''(h) = 2A_1\Phi + A_1 h \frac{d\Phi}{dh}$$

$$\Delta(0) = 0;$$

$$\Delta'(0) = (A_0 + A_1) f$$

$$\Delta''(0) = 2A_1\Phi = 2A_1\{\alpha_1 f_x + \beta_{10} f_y f\}.$$

Порівнюючи (6.5) і (6.7)

$$y_n + hf + \frac{h^2}{2}(f_x + f_y f) = y_n + h(A_0 + A_1) + h^2 A_1(\alpha_1 f_x + \beta_{10} f_y f)$$

отримаємо $A_0 + A_1 = 1, \ \alpha_1 A_1 = \frac{1}{2}, \ \beta_{10} A_1 = \frac{1}{2}$

Починаючи з $q > 1$ характерною є ситуація коли число невідомих більше числа рівнянь для їх визначення. Тобто A, α, β неможливо знайти однозначно. Тоді звичайно одну з невідомих приймають як незалежну змінну. Якщо за незалежну змінну взяти у даному випадку A_1, то отримаємо

$$A_0 = 1 - A_1, \ \alpha_1 = \frac{1}{2A_1}. \qquad \beta_{10} = \frac{1}{2A_1}$$

Конкретні значення A_1 вибирають з міркувань простоти або симетрії робочих формул. Якщо прийняти, що $A_1 = 1$, то $A_0 = 0; \ \alpha_1 = \beta_{10} = \frac{1}{2}$ і отримаємо формулу Рунге другого порядку

$$\Delta = hf(x_n + \frac{h}{2}, y_n + \frac{h}{2} f) \qquad (6.10)$$

При $A_1 = \frac{1}{2}$ маємо $A_0 = \frac{1}{2}$; $\alpha_1 = \beta_{10} = 1$, що приводить до іншої формули цього ж порядку

$$\Delta = \frac{1}{2}(\varphi_0 + \varphi_1) \;\; \varphi_1 = hf(x_n + h, y_n + \varphi_0) \qquad (6.11)$$

Для найбільш поширеного метода Рунге-Кутти четвертого порядку маємо

$$\begin{aligned} y_{n+1} &= y_n + \frac{1}{6}(\varphi_0 + 2\varphi_1 + 2\varphi_2 + \varphi_3) \\ \varphi_0 &= hf(x_n, y_n), \\ \varphi_1 &= hf(x_n + \frac{h}{2}, y_n + \frac{\varphi_0}{2}), \\ \varphi_2 &= hf(x_n + \frac{h}{2}, y_n + \frac{\varphi_1}{2}), \\ \varphi_3 &= hf(x_n + h, y_n + \varphi_2). \end{aligned} \qquad (6.12)$$

Класичній метод Рунге-Кутти можна розглядати як узагальнення на диференціальні рівняння квадратурної формули Сімпсона

$$\int_{x_n}^{x_{n+1}} f(x)dx \approx \frac{h}{6}\left[f(x_n) + 4\left(\frac{f(x_n) + f(x_{n+1})}{2} \right) + f(x_{n+1}) \right]$$

Якщо $f(y, x)$ є функція тільки від x , то обидві формули співпадають.

До переваг методи Рунге-Кутти слід віднести їх чисельну і стійкість і легкість програмної реалізації .

Методи типа Рунге відносяться до одно крокових (самостартуючих), постільки для обчислення функції в наступній точці досить мати її значення у попередній. Так як, для методів одного порядку точність рішення залежить від

кроку h це дає можливість проводити обчислення з заданою точністю. Наприклад критерієм зменшення величини кроку в два рази є виконання умови

$$\left| y_n\left(\frac{h}{2}\right) - y_n(h) \right| \frac{\alpha}{\alpha - 1} > \varepsilon, \qquad \alpha = 2^k,$$

де k N-порядок методу.

Багатокрокові методи. В методах, які розглядалися досі, значення y_{n+1} обчислюються за допомогою функції що, залежить від. y_n і довжини кроку h_n тільки в одній попередній точці x_n. Мабуть, буде логічно припустити , що можна отримати більшу точність, якщо використати інформацію з декількох попередніх точках, , а саме $y_n, y_{n-1}, \ldots, y_{n-k}$ і $f_n, f_{n-1}, \ldots, f_{n-k}$. Багатокрокові методи, в в основу яких закладена ця ідея є досить ефективними. Якщо потрібна висока точність, то вони звичайно більш економічні, ніж одно крокові, і часто можна тривіально отримати оцінку похибки усічення. Запрограмовані відповідним чином, багатокрокові методи можуть ефективно видавати значення чисельного розв'язку в довільних точках, не змінюючи корку h. Порядок метода може обиратися автоматично і динамічно змінюватися, тим самим виходять методи, що працюють для дуже широкого кола задач. Ці переваги досягаються ціною ускладнення програми і в деяких випадках за рахунок високої чисельної нестійкості.

Лінійні багатокрокові методи можна розглядати як спеціальні випадки формули

$$y_{n+1} = \sum_{i=1}^{k} \alpha_i y_{n+1-i} + h \sum_{i=0}^{k} \beta_i f_{n+1-i}$$

Де h - фіксоване ціле число і α_q або β_q відмінно від нуля. Ці формула визначає загальний лінійний k - кроковий метод. Метод називається лінійним,

тому що кожне f_i входить в формулу лінійно, при цьому f_i може бути як лінійною так і нелінійною функцією своїх аргументів.

Після «старту» методу кожен крок потребує обчислення y_{k+1} по відомим значенням $y_n, y_{n-1},...,y_{n-k+1}, f_n, f_{n-1},...,f_{n-k+1}$. Якщо $\beta_0 = 0$, то метод називається явним і обчислення проводяться очевидним чином. Якщо ж $\beta_0 \neq 0$, то метод називається неявним, тому що для знаходження y_{n+1} треба мати f_{n+1}. Збільшення труднощів при використанні неявних методів окупається іншим їх якостями.

Звичайно на кожному кроці розв'язку використовуються два багатокрокові методи. Явний метод називається прогнозом , і супроводжується одним чи більше застосуванням неявного методу, який називається корекцією - звідси і назва «методи прогнозу і корекції». Досить часто ці методи називаються ще методами « предиктор – коректором».

Отримаємо робочі формули для багатокрокових методів.

Згідно (6.6) розв'язок задачі Коші може бути записано у формі

$$y_{n+1} = y_k + \int_{x_n}^{x_n+h} f(x,y)dx \quad y_{n+1} = y_n + \Delta_n$$

Введемо для додатку Δ_n позначення Δ_P і Δ_C для кроків прогнозу і корекції відповідно. Будемо вважати, що розв'язок відомий у всіх попередніх точках до x_{n-k} точки включно і вузли розташовані рівномірно з кроком h, що дозволяє по цим значенням побудувати інтерполяційний поліном k- го ступеня. Запишемо інтерполяційний поліном Ньютона у вигляді

$$f(x) = f_n + (x - x_n)\nabla f_n + (x - x_n)(x - x_{n-1})\nabla^2 f_n + ...,$$

де $\nabla^m f_n$- скінченні різниця, порядку m при інтерполюванні «назад». Для зручності процедури інтегрування введемо нову змінну $t = \frac{x - x_n}{h}$. Тоді $x_i = x_n - (n-i)h$, $x - x_i = (t + n - i)h,$ а підінтегральна функція набуде вигляду

$$f(t) = f_n + t\nabla f_n + \frac{t(t+1)}{2!}\nabla^2 f_n + \ldots$$

Після інтегрування маємо наступний результат

$$\begin{aligned} \Delta_P = \int_0^1 f(t)dt = h\int_0^1 \left[f_n + t\nabla f_n + \frac{t(t+1)}{2!}\nabla^2 f_n + \ldots \right] dt = \\ = h\left[f_n + \frac{1}{2}\nabla f_n + \frac{1}{12}\nabla^2 f_n + \ldots \right] \end{aligned} \qquad (6.13)$$

При $k = 0$ підінтегральний вираз в (6.13) є константою і в результаті маємо звичайний метод Ейлера. Випадок $k = 1$ відповідає лінійній інтерполяції, і після елементарних обчислень маємо формулу прогнозу першого порядку

$$y_{n+1} = y_n + \Delta_P = y_n + \frac{h}{2}(3f_n - f_{n-1})\ldots$$

Отримане значення $y_{n+1} = y_n + \Delta_P$ можна використати для обчислення $f_{n+1} = f(x_{n+1}, y_{n+1})$ і за його допомогою провести уточнення (корекцію) функції на під інтервалі $[x_n; x_{n+1}]$. Для обчислення інтегралу на кроці корекції введемо позначення

$$t = \frac{x - x_{n+1}}{h} \quad x = x_n,\ t = -1 \ \ x = x_{n+1}, t = 0$$

а для інтерполяційного поліному маємо

$$f(t) = f_{n+1} + t\nabla f_{n+1} + \frac{t(t+1)}{2!}\nabla^2 f_{n+1} + \ldots$$

Після інтегрування отримуємо

$$\Delta_C = \int_0^1 f(t)dt == h\left[f_{n+1} - \frac{1}{2}\nabla f_{n+1} + \frac{1}{12}\nabla^2 f_{n+1} + \ldots \right] \qquad (6.14)$$

Використовуючи формули (6.13) і (6.14) можна отримати формули будь якого порядку. При цьому на кроці прогнозу і на кроці корекції порядок цих формул може бути різним.

На практиці самими розповсюдженим є методи прогнозу - корекції Адамса четвертого порядку:

крок прогнозу $y_{n+1} = y_n + \frac{h}{24}(55 f_n - 59 f_{n-1} + 37 f_{n-2} - 9 f_{n-3})$

крок корекції $y_{n+1} = y_n + \frac{h}{24}(9 f_{n+1} + 19 f_n - 5 f_{n-1} + f_{n-2})$

Обчислення y_{n+1} по схемі прогноз-корекція проводяться наступним чином. Знаходяться значення Δ_P, $\Delta_C^{(0)}$, номеру ітерації присвоюється значення $i = 0$ і задається величина похибки ε. Якщо $\left|\Delta_P - \Delta_C^{(0)}\right| \le \varepsilon$, то приймається $y_{n+1} = y_n + \Delta_P$. Якщо ця умова не виконується, то проводиться наступний ітераційний процес:

Початок. Послідовно обчислюються величини: $y_{n+1} = y_n + \Delta_C^{(i)}$, f_{n+1}, покладається $i = i + 1$ та знаходиться $\Delta_C^{(i)}$. Якщо виконується умова $\left|\Delta_C^{(i)} - \Delta_C^{(i-1)}\right| \le \varepsilon$ то $y_{n+1} = y_n + \Delta_C^{(i)}$ і розв'язок знайдено. Якщо ні то повертаємося до *початку* процесу.

На практиці для досягнення збіжності достатньо не більш двох ітерацій.

6.3 Похибки чисельного розв'язку задачі Коші

Існує два джерела похибок при чисельному розв'язку задачі Коші:

- похибка дискретизації(похибка усічення)
- похибка округлення.

Похибка дискретизації є властивість методу, що застосовується. Це значить, що якби всі арифметичні обчислення могли виконуватися з нескінченною точністю, то інших похибок, окрім похибка дискретизації, не було б. Похибка округлення є властивість машини і програми. Оскільки точний розв'язок, взагалі кажучи, невідомий і не може бути обчисленим, то звичайно тим чи іншим чином оцінюється похибка дискретизації. Важливо розрізняти між собою дві міри похибки дискретизації: *локальну* і *глобальну* . Локальну похибка дискретизації – це похибка зроблена на поточному кроці, при умові, що попередні значення є точними і що відсутні помилки округлення. Більш точно, нехай $u_n(t)$- функція від t, яка описується умовами

$$u_n' = f(t, u_n)$$

$$u_n = y_n$$

Таким чином $u_n(t)$ - розв'язок диференціального рівняння, що визначається не початковою умовою в точці t_0, а значенням обчисленого розв'язку у точці t_n. Локальною похибкою дискретизації є

$$d_n = y_{n+1} - u_n(t_{n+1})$$

Це різниця між обчисленим розв'язком і теоретичним розв'язком, визначеними одними й тими ж даними у точці t_n.

Глобальною похибкою дискретизації є різниця між обчисленим розв'язком і теоретичним розв'язком, визначеним початковою умовою у точці t_0, тобто

$$e_n = y_n - y(t_n).$$

Різницю між локальною і глобальною похибками дискретизації легко бачити в спеціальному випадку, коли $f(t,y)$ не залежить від y. У цьому випадку розв'язком є просто інтеграл $y=\int_{t_0}^{t} f(\tau)d\tau$. Метод Ейлера у цьому випадку перетворюється у схему чисельного інтегрування , в так звану «формулу прямокутників для лінивих», яка використовує значення функції на кінцях підінтервалів, а не в середніх точках:

$$\int_{t_0}^{t_N} f(\tau)d\tau \approx \sum_{n=0}^{N-1} h_n f(t_n).$$

Локальна похибка дискретизації є похибка на одному під інтервалі

$$d_n = h_n f(t_n) - \int_{t_n}^{t_{n+1}} f(\tau)d\tau,$$

а глобальна похибка дискретизації - загальна похибка

$$e_N = \sum_{n=0}^{N-1} h_n f(t_n) - \int_{t_0}^{t_N} f(\tau)d\tau$$

В даному спеціальному випадку кожен під інтеграл є незалежним від інших (сума може обчислюватися в довільному порядку), так що глобальна похибка є сумою локальних похибок:

$$e_N = \sum_{n=0}^{N-1} d_n$$

У загальному, випадку коли права частина диференціального рівняння має вигляд $f(t,y)$, похибка на будь якому інтервалі залежить від рішень, обчислених для попередніх інтервалів. Внаслідок цього глобальна похибка в загальному випадку буде більше суми локальних похибок, якщо диференціальне рівняння є нестійким, але менше цієї суми , якщо диференціальне рівняння є стійким.

6.4 Розв’язок крайових задач

Розглянемо крайову задачу

$$y'' = F(x, y, y') \tag{6.15}$$

$$y(a) = \alpha, \qquad y(b) = \beta \tag{6.16}$$

У випадку лінійної задачі будь-яка лінійна комбінація часткових рішень диференціального рівняння є також його рішенням. Розв’язок задачі шукається у вигляді

$$y(x) = k_1 y_1(x) + k_2 y_2(x)$$

Коефіцієнти k_1 і k_2 визначаються шляхом розв’язку двох задач Коші для рівняння (6.15) з початковими умовами

$$y_1(a) = \alpha, \quad y_1'(a) = \gamma_1 \tag{6.17}$$

$$y_2(a) = \alpha, \quad y_2'(a) = \gamma_2 \tag{6.18}$$

Якщо отримані рішення у точці $x = b$ мають значення $y_1(b) = \beta_1$ і $y_2(b) = \beta_2$, то умова (6.16) утворює систему алгебраїчних рівнянь відносно невідомих k_1 і k_2

$$k_1 + k_2 = 1$$

$$k_1\beta_1 + k_2\beta_2 = \beta$$

І остаточне рішення лінійної крайової задачі має вигляд

$$y(x) = \frac{1}{\beta_1 - \beta_2}\left[(\beta - \beta_2)y_1(x) + (\beta_1 - \beta)y_2(x)\right]$$

Для рішення нелінійної задачі звичайно використовується метод пострілів, у якому крайова задача зводиться до послідовності задач Коші. Знайдемо розв'язок двох задач Коші для рівняння (6.15), у яких друга початкова умова має вигляд $y_1'(a) = \alpha_1$ і $y_2'(a) = \alpha_2$, а рішення при $x = b$ відповідно $y_1(b) = \beta_1$ $y_2(b) = \beta_2$. Завжди можна підібрати такі значення α_1 і α_2, щоб виконувалась умова

$$(\beta - \beta_1)(\beta - \beta_2) \le 0 \ ,$$

тобто α_1 і α_2 утворюють інтервал невизначеності для нелінійного рівняння

$$y(\alpha_k, b) - \beta = 0 \ ,$$

де y - рішення задачі Коші

$$y'' = F(x, y, y');$$
$$y(a) = \alpha; \qquad y'(a) = \alpha_k$$

Досить поширеним підходом до рішення лінійних крайових задач є зведення диференціального рівняння до системи алгебраїчних за допомогою скінчено-різницевої апроксимації похідних. Якщо у крайовій задачі

$$y'' + p(x)y' + g(x)y = z(x)$$
$$y(a) = \alpha; \qquad y(b) = \beta$$

замінити похідні за допомогою формул чисельного диференціювання, то отримуємо наступну СЛАР

$$(2 - p_i h)y_{i-1} + (2h^2 g_i - 4h)y_i + (2 + p_i h)y_{i+1} = 2h^2 z_i, \quad \text{i} = 2,3,\ldots,\text{n} - 1$$
$$y_1 = \alpha, \quad y_{\text{n}} = \beta$$

де прийняті позначення

$$\text{y}_i = y(x_i), \quad p_i = p(x_i), \quad g_i = g(x_i), \quad z_i = z(x_i) \ .$$

Постільки матриця системи має трьох діагональну структуру, то для її розв’язку доцільно застосувати метод прогонки.

РОЗДІЛ 7 РІВНЯННЯ З ЧАСТКОВИМИ ПОХІДНИМ

До рівнянь з частковими похідними приводять задачі механіки, теплопровідності, переносу випромінювання, квантової механіки і багато інших. Незалежними змінними у фізичних задачах звичайно є час t і координати r . Повна математична постановка містить диференціальне рівняння , а також додаткові умови, які дозволяють виділити його єдиний розв'язок. Додаткові умови звичайно задаються на границі області Ω, в якій цей розв'язок шукається.

7.1 Класифікація рівнянь та методів їх рішень

У загальному випадку диференціальні рівняння другого порядку зі змінним коефіцієнтами мають вигляд

$$a_{11}(x,y)\frac{\partial^2\varphi}{\partial x^2}+a_{12}(x,y)\frac{\partial^2\varphi}{\partial x\partial x}+a_{22}(x,y)\frac{\partial^2\varphi}{\partial y^2}=0$$

В залежності від знаку величини $D=a_{11}a_{22}-a_{12}^2$ у деякій точці вони поділяються на наступні типи

1) $D<0$ – еліптичний тип. Наприклад

$$\frac{\partial^2\Phi}{\partial x^2}+\frac{\partial^2\Phi}{\partial y^2}=0$$ (рівняння Лапласа)

Ці рівняння описують стан рівноваги суцільних середовищ.

2) $D=0$ – рівняння параболічного типу. Наприклад

$$\frac{\partial\Phi}{\partial t}=\frac{\partial^2\Phi}{\partial x^2}$$

або

$$\frac{\partial \Phi}{\partial t} = \frac{\partial^2 \Phi}{\partial x^2} + \frac{\partial^2 \Phi}{\partial y^2} = \nabla^2 \Phi.$$

Рівняння параболічного типу описують процеси теплопровідності і дифузії.

3) $D > 0$ - рівняння гіперболічного типу

$$\frac{\partial^2 \Phi}{\partial t^2} = \frac{\partial^2 \Phi}{\partial x^2} \quad \text{або} \quad \frac{\partial^2 \Phi}{\partial t^2} = \frac{\partial^2 \Phi}{\partial x^2} + \frac{\partial^2 \Phi}{\partial y^2} = \nabla^2 \Phi$$

Гіперболічні рівняння мають ще назву хвильових ,так як описують коливальні процеси. Приведенні вище рівняння мають загальну назву – рівняння математичного фізики.

Процес рішення рівнянь математичної фізики (РМФ) складається з наступних етапів.

Вибір методу рішення. Найбільш поширеними з існуючих методів є метод скінчених елементів, метод різниць і розкладання рішення по базисних функціях.

Дискретизація області рішення. На цьому етапі неперервна область зображується у вигляді множини дискретних точок (за винятком метода скінчених елементів). Така множина звичайно називається сіткою і її форма залежить, як правило, від системи координат, в якій записані рівняння. Так сітки (рис 7.1) і (рис 7.2) відповідно використовуються для полярної і декартової систем координат.

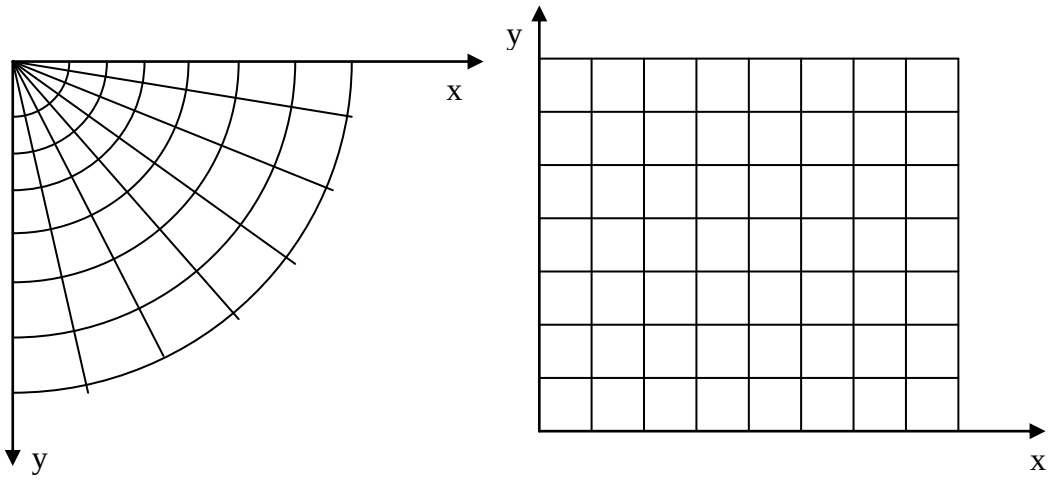


Рис. 7.1.- Дискретизація в полярних координатах.

Рис. 7.2.- Дискретизація в декартових координатах.

Побудова обчислювального шаблона. Обчислювальний шаблон визначається тими точками сітки, які використовуються для знаходження чисельного значення похідних.

Рішення СЛАР. Вибір методу рішення залежить від структури матриці системи: суцільно заповненої, діагонально – орієнтованої, блочної, розрідженої.

7.2 Рівняння параболічного типу

До цього типу належить рівняння

$$\frac{\partial U}{\partial t} = D\frac{\partial^2 U}{\partial x^2} \quad , \qquad (7.2.1)$$

D - Коефіцієнт дифузії. Це рівняння є математичною моделлю процесу дифузії в стержні. При рішенні фізичних (технічних) задач загальноприйнятим є перехід до безрозмірних змінних. Такий підхід дає можливість зменшити похибку обчислювань, полегшити аналіз результатів, а також дає змогу для порівняння результатів одержаних різними методами. У даному рівнянні t має розмірність часу, а x – розмірність довжини. Якщо l – довжина стержня, то перехід до безрозмірної координати x проводиться наступним чином

$$\bar{x} = \frac{x}{l}$$

Тоді $$\frac{\partial U}{\partial x} = \frac{\partial U}{\partial \bar{x}}\frac{\partial \bar{x}}{\partial l} = \frac{1}{l}\frac{\partial U}{\partial \bar{x}},$$

$$\frac{\partial^2 U}{\partial x^2} = \frac{1}{l^2}\frac{\partial^2 U}{\partial \bar{x}^2}.$$

Якщо ввести безрозмірний час, визначений як $\tau = kt,\ k$ – деяка невизначена поки константа, то $\frac{\partial U}{\partial t} = k\frac{\partial U}{\partial \tau}$ і рівняння (7.2.1) набуває вигляду

$$k\frac{\partial U}{\partial \tau}=\frac{1}{l^2}\frac{\partial^2 U}{\partial \bar{x}^2}. \qquad (7.2.2)$$

Константа k визначається з вимоги, щоб у остаточній формі у рівняння не входили параметри з розмірністю. Тоді якщо у (7.2.2) покласти $k=\frac{D}{l^2}$, то вихідне рівняння набуває вигляду

$$\frac{\partial U}{\partial \tau}=\frac{\partial^2 U}{\partial x^2} \qquad (7.2.3)$$

Далі для спрощення позначень замість $\bar{x}$ буде використовуватись x. Аналогічним чином можна показати, що до вигляду (7.2.3) зводиться рівняння теплопровідності

$$\frac{\partial U}{\partial t}=\frac{\lambda}{c\rho}\frac{\partial^2 U}{\partial x^2}$$

де λ коефіцієнт теплопровідності, c - питома теплоємність, ρ-щільність.

Тобто два фізичні процеси у безрозмірних змінних описуються одним і тим же рівнянням.

Таким чином необхідно знайти функцію $U(\tau,x)$, що задовольняє рівнянню

$$\frac{\partial U}{\partial \tau}=\frac{\partial^2 U}{\partial x^2}$$

початковій $U(0,x)=\varphi(x)$

та граничним умовам $U(\tau,0)=f_1(\tau), U(\tau,1)=f_2(\tau)$

Область, в якій шукається рішення є прямокутною смугою розміри якої визначаються граничними та початковою умовами, тобто

$$0 \le x \le 1\ ,\ 0 \le \tau \le \tau_e .$$

Величина τ_t є граничним значенням проміжку часу до якого ведеться пошук Найбільш доцільним є дискретизація області прямокутною сіткою

$$\begin{cases} x_i = (i-1)h, i = 1,2,\ldots,n_x \\ \tau^{\mathrm{j}} = (j-1)\Delta\tau,\ j = 1,2,..n_y \end{cases},$$

де параметри h і $\Delta\tau$ величина кроку по координаті та часу вибираються з міркувань точності наближення рішення і обсягу обчислень. На рис. 7.3 показана область дискретизації і помічена точка з координатами $\left(x_i, \tau^{\,j}\right)$,у якій функція U приймає значення U_i^j . Параметри сітки h і $\Delta\tau$, величина кроку по змінним x і τ , відповідно, вибираються з міркувань точності наближення рішень і обсягу обчислень.

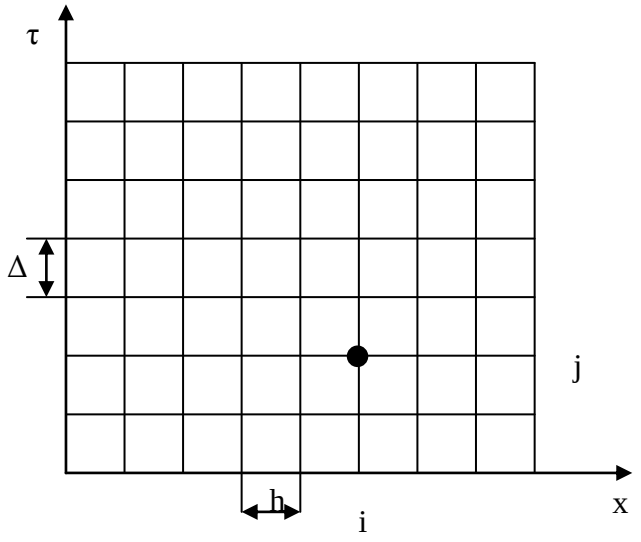


Рис. 7.3.- Дискретизація задачі теплопровідності

Після заміни похідних у (7.2.3) їх скінчено різницевими аналогами по формулам чисельного диференціювання маємо

$$\frac{U_i^{j+1} - U_i^j}{\Delta\tau} = \frac{U_{i+1}^k - 2U_i^k + U_{i-1}^k}{h^2} \qquad (7.2.4)$$

Де k – номер часового шару, для якого обчислюється похідна по x. В залежності від значення к отримуємо різні обчислювальні схеми. Так при $k = j$ похідна обчислюється на попередньому часовому шарі і (7.2.4) набуває вигляду

$$\frac{U_i^{j+1} - U_i^j}{\Delta\tau} = \frac{1}{h^2}\left(U_{i+1}^j - 2U_i^j + U_{i-1}^j\right) \quad (7.2.5$$

Рівняння (7.2.5) має тільки одну тільки одну невідому U_i^{j+1}, яка може бути знайдена у явній формі і тому ця схема називається явною різницею схемою. По цій схемі маємо

$$U_i^{j+1} = U_i^j + \frac{\Delta\tau}{h^2}\left(U_{i+1}^j - 2U_i^j + U_{i-1}^j\right) \quad (7.2.6)$$

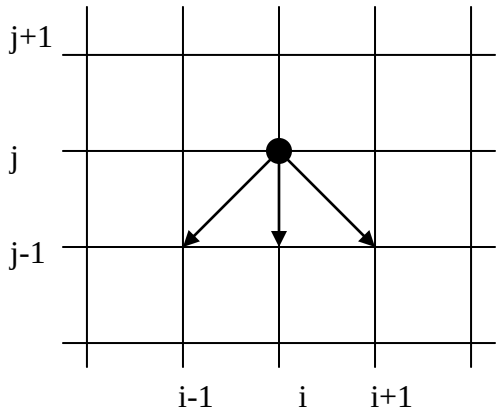


Рис. 7.4.- Обчислювальний шаблон явної різницевої схеми

Обчислення по формулі (7.2.6) починається з $j = 1$, так як при $j = 0$ рішення відоме з початкової умови. Параметр i змінюються від $i = 0$ до $i = n_x$

Значення функції при $i = 0$ та $i = n_x$ або вже відоме з граничних умов, якщо на границі задана сама функція, або може бути знайдене за допомогою формул

чисельного диференціювання, якщо на якійсь з границь дана похідна. Наприклад якщо гранична умова має вид

$$\frac{\partial U}{\partial x}\Big|_{x=0} = F(\tau),$$

що аналогічно умові $\frac{U_1^j - U_0^j}{h} = F(\tau)$ то

$$U_0^j = U_1^j - hF(\tau).$$

Співвідношення сторін сітки визначається величиною параметра $r = \frac{\Delta\tau}{h^2}$, який є мірою стійкості схеми. Схема (7.2.6) є чисельно нестійкою, тобто швидко накопичує обчислювальну похибку. Для забезпечення її стійкості необхідно виконання умови

$$r = \frac{\Delta\tau}{h^2} \le \frac{1}{2} \qquad (7.2.7)$$

Якщо r вибрати з умови $r = \frac{1}{2}$, то формула (7.2.6) набуває дуже просту форму

$$U_i^{j+1} = \frac{1}{2}\left(U_{i-1}^j + U_{i+1}^j\right).$$

Але визначення параметру r з умови (7.2.7) приводить до досить малих величин кроку $\Delta\tau$, що обумовлює значну обчислювальну похибку і тому явну різницеву схему доцільно використовувати для відносно невеликого значення параметру τ_e. Якщо в (7.2.4)похідна n0 x обчислюється поточному часовому шарі $(k = j+1)$, то отримане рівняння уже містить три невідомі $U_{i-1}^{j+1}, U_i^{j+1}, U_{i+1}^{j+1}$, які знайти у явній формі з одного рівняння неможливо, і тому ця схема називається неявної скінчено. Шаблон такої скінчено різницевої схеми має вигляд (рис. 7.5)

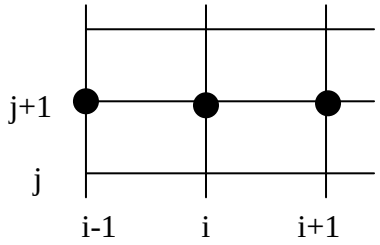


Рис 7.5.- Обчислювальний шаблон неявної різницевої схеми

У цьому випадку задача зводиться до розв'язку СЛАР

$$rU_{i-1}^{j+1} + (1-2r)U_i^{j+1} - rU_{i+1}^{j+1} = U_i^j$$

Матриця якої має трьох діагональну структуру. На відмінну від явної схеми неявна кінчено – різницева схема є стійкою і ніяких обмежень на параметр r не потребує.

У випадку коли $U = U(x, y)$ маємо рівняння

$$\frac{\partial U}{\partial \tau} = \frac{\partial^2 U}{\partial x^2} + \frac{\partial^2 U}{\partial y^2} = \nabla^2 U. \qquad (7.2.8)$$

У скінчено – різницевій формі (7.2.8) запишеться у вигляді

$$\frac{U_{ij}^{k+1} - U_{ij}^k}{\Delta \tau} = L_x^m + L_y^m \quad (7.2.9)$$

Індекси i, j, k у (7.2.9) відповідають руху по змінним x, y та τ. Скінчено різницевий оператор L_x^p має вигляд

$$L_x^p = \frac{U_{i+1}^p - 2U_i^p + U_{i-1}^p}{h_x^2}$$

Аналогічно форму має оператор L_y^p. Якщо у рівняння (7.2.9) покласти $m = p = k$, то одержимо явну різницеву схему для випадку області з двома геометричними вимірами, що приводить до рішення

$$U_{ij}^{k+1} = U_{ij}^{k} + \Delta\tau\left(L_x^k + L_y^k\right)$$

Як і у випадку (7.2.5) ця схема є нестійкою, а накопичення похибки відбувається ще швидше. При $m = p = k+1$ рівняння (7.2.9) приводить до неявної різницевої схеми. Ця схема хоч і є стійкою, але рішення СЛАР, до якої зводиться задача, потребує дуже великих обсягів обчислень. Дійсно, якщо взяти сітку з гранично допустимою для чисельного диференціювання з кроком $N_x = N_y = 0{,}1$, звичайно після нормування області, то матриця системи має досить великий розмір, а саме 121×121 $\left(N_x = N_y = 11\right)$ Тому на границі використовується більш економічний метод, який має назву Метода змінних напрямків. У цьому методі рішення отримується за два етапи. На першому етапі з системи рівнянь

$$\frac{W_{ij}^{k+1} - U_{ij}^{k}}{\Delta\tau} = L_x^{k+1} + L_y^k$$

знаходиться значення допоміжної функції $W_{ij}^{k=1}$. На другому етапі уже безпосередньо визначається функція U_{ij}^{k+1} . Для цього треба розв'язати систему

$$\frac{U_{ij}^{k+1} - U_{ij}^{k}}{\nabla\tau} = L_x^{k+1}(W) + L_y^{k+1}(U)$$

Таким чином спочатку знаходиться рішення одновимірної по координаті x задачі, далі обчислюється $L_x^{k+1}(W)$, і на кінець визначається рішення одновимірної задачі в напрямку по y.

7.3 Рівняння гіперболічного типу

Рішення рівнянь гіперболічного типу буде розглянуто на прикладі рівняння коливання струни

$$\frac{\partial U}{\partial\tau^2} = \frac{\partial^2 U}{\partial x^2} \qquad (7.3.1)$$

з граничними умовами

$$U(\tau,0) = f_1(\tau),$$

$$U(\tau,1) = f_2(\tau)$$

і двома початковими умовами, а саме

$$U(0,x) = \varphi_1(\tau)\ , \qquad \left.\frac{\partial U(\tau,x)}{\partial \tau}\right|_{\tau=0} = \varphi_2(x)$$

Які у випадку рівняння параболічного типу використаємо скінчено різницевий підхід. Це зовсім не означає, що не існує інших методів рішення цих задач. Так наприклад, для рівняння параболічного типу рішення можна провести методом становлення, а для гіперболічних рівнянь методами, що в основі мають перетворення Лапласа. Але скінчено – різницеві схеми є найбільш поширеними, по скільки вони відносно прості, найбільш вивчені і тому при їх використанні можна оцінити надійність і точність отриманих результатів.

У скінчено – різницевій формі рівняння (7.3.1) запишеться як

$$\frac{U_i^{j+1} - 2U_i^j + U_i^{j-1}}{\Delta\tau^2} = \frac{U_{i+1}^j - 2U_i^j + U_{i-1}^j}{h^2} \qquad (7.3.2)$$

де параметри i, j, h та $\Delta\tau$ мають той же зміст . З (7.3.2) безпосередньо знаходиться

$$U_i^{j+1} = \gamma^2 U_{i-1}^j + 2(1-\gamma^2 U_i^j) + \gamma^2 U_{i+1}^j - U_i^{j-1} \qquad (7.3.3)$$

Рішення (7.3.3) по змісту є аналогічним явній різницевій схемі і його стійкість також визначається величиною параметра r . При $r > 1$ скінчено – різницева апроксимація є нестійкою, при $r = 1$ стійкою і наближене рішення співпадає з точним, а при $r < 1$ обчислювальний процес хоч і є стійким, але його точність зменшується по мірі зменшення величини r .

7.4 Еліптичні рівняння

Ррозглянемо декілька методів рішення рівнянь еліптичного типу:

$$\frac{\partial^2 U}{\partial x^2} + \frac{\partial U^2}{\partial y^2} = f(x, y) \qquad (7.4.1)$$

В області Ω з границею Γ, на якій функція $U(x, y)$ приймає значення $U_0(x, y)$. Традиційно почнемо з скінчено – різницевої апроксимації. Пошук рішення зводиться до визначення значень $U(x, y))$, тобто величини $U_{ij} = U(x_i, y_i)$ у вузлах прямокутної сітки

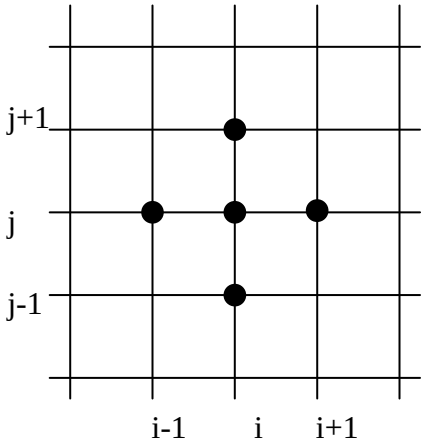


Рис. 7.6.- Обчислювальний шаблон «хрест»
для еліптичних і гіперболічних задач

Обчислювальний шаблон для визначення похідних по формулах чисельного диференціювання показано на рис. 7.6. У кожній точці Ω за винятком граничних (7.4.1) набуває вигляду

$$\frac{U_{i+1,j} - 2U_{ij} + U_{i-1,j}}{hx^2} + \frac{U_{i,j+1} - 2U_{ij} + U_{i,j-1}}{hy^2} = f_{ij} \ , \qquad (7.4.2)$$

де h_x, h_y, параметри сітки (7.2.4) відповідно по x і y координатам, а $f_{ij} = f(x_i, y_j)$. Міняючи індекси i, j в (7.2.4), так щоб пройти всі внутрішні точки Ω отримуємо СЛАР відносно невідомих U_{ij}. Матриця

цієї системи має п’яти діагональну форму і для розв’язку СЛАР, як правило використовується ітераційний метод Зейделя.

Один із варіантів метода Зейделя у випадку коли $h_x=h_y=h$ має вигляд

$$U_{ij}^{k+1}=\frac{1}{4}\left[h^2 f_{ij}-U_{i+1,j}^{k}-U_{i-1,j}^{k}-U_{i,j+1}^{k}-U_{i,j-1}^{k}\right]$$

Де k – номер ітерації. Визначення рішення задачі (7.4.1) шляхом розкладу його по базисним функціям

h_x

$$U(x,y)=\sum_{i=0}^{n}\alpha_i\psi_i(x,y) \qquad (7.4.3)$$

є особливо зручним коли $f(x,y)=0$, тобто для задачі

$$\nabla^2 U(x,y)=0,\quad U|_{\Gamma}=U_0.$$

Для цього використовують базисні функції, які задовольняють умові

$$\nabla^2\psi_i(x,y)=0$$

І тоді задача складається у визначенні коефіцієнтів в α_i у (7.4.3) при яких виконується гранична умова.

У якості таких функцій можна взяти :

У випадку $U=U(x,y)$ функцію $\psi_i=r^i T_i(t)$

Де $r=\sqrt{x^2+y^2}$, $t=\cos\frac{x}{r}$ або $t=\cos\frac{y}{r}$

$T_i(t)$ – поліноми Чебишева.

Для випадку трьох вимірної задачі $U=U(x,y,z)$

Функцію $\psi_i=\rho^i P_i(t)$

де $\rho = \sqrt{x^2 + y^2 + z^2}$, $t = \cos\frac{x}{\rho}$, $t = \cos\frac{y}{\rho}, t = \cos\frac{z}{\rho}$, а

$P_i(t)$ – поліноми Лежандра.

У випадку коли рішення шукається у циліндричній системі координат $U = U(r, z, \varphi)$ і воно задовольняє умові осьової симетрії по координаті φ то

$$\psi_i(r, z) = \rho^i P_i(t)$$

де $t = \cos\frac{t}{\rho}$, $\rho = \sqrt{r^2 + z^2}$

Самим же загальним підходом до рішення рівнянь математичної фізики варіаційна постановка задачі. Основні ідеї цього підходу, а також деякі методи його чисельної реалізації будуть розглянуті на прикладі рівнянь еліптичного типу.

7.5 Варіаційний підхід до рішення рівнянь у часткових похідних

Розглядається задача рішення рівняння еліптичного типу, а саме рівняння Лапласа в області Ω границею Γ при даній граничній умові

$$\begin{cases} \dfrac{\partial^2 U}{\partial x^2} + \dfrac{\partial^2 U}{\partial y^2} = 0 \\ U\big|_\Gamma = U_0 \end{cases}$$

що має назву задачі Діріхле. Нехай U є функція, що задовольняє як рівнянню, так і умові на границі Γ. таке рішення називається дійсним. Функція V, яка задовольняє умові на границі $V\big|_\Gamma = U_0$ і необов'язково рівнянню, називається допустимим рішенням або пробною функцією.

Якщо в очевидних співвідношеннях

$$\frac{\partial}{\partial x}\left[\frac{\partial U}{\partial x}(V-U)\right]=\frac{\partial^2 U}{\partial x^2}(V-U)+\frac{\partial U}{\partial x}\left(\frac{\partial V}{\partial x}-\frac{\partial U}{\partial x}\right)$$

$$\frac{\partial}{\partial y}\left[\frac{\partial U}{\partial y}(V-U)\right]=\frac{\partial^2 U}{\partial y^2}(V-U)+\frac{\partial U}{\partial y}\left(\frac{\partial V}{\partial y}-\frac{\partial U}{\partial y}\right)$$

підсумувати відповідно ліві і праві частини і результат про інтегрувати по області Ω, то

$$\int_\Omega\left(\frac{\partial^2 U}{\partial x^2}+\frac{\partial^2 U}{\partial y^2}\right)d\Omega=\int_\Omega\left\{\left[\frac{\partial}{\partial x}\left(\frac{\partial U}{\partial x}(V-U)\right)\right]+\left[\frac{\partial}{\partial y}\left(\frac{\partial U}{\partial y}(V-U)\right)\right]-\right.$$

$$\left.-\frac{\partial U}{\partial x}\left(\frac{\partial V}{\partial x}-\frac{\partial U}{\partial x}\right)-\frac{\partial U}{\partial y}\left(\frac{\partial V}{\partial y}-\frac{\partial U}{\partial y}\right)\right\}d\Omega \qquad (7.5.1)$$

Так як U є дійсним рішенням, то ліва частина в (7.5.1) рівна нулю, а права за допомогою теореми Гауса – Остроградського приводиться до вигляду

$$\int_\Gamma\left[\frac{\partial U}{\partial x}(V-U\cos(\nu,x))+\frac{\partial U}{\partial y}(V-U\cos(\nu,y))\right]d\Gamma-$$

$$-\int_\Omega\left[\frac{\partial U}{\partial x}\left(\frac{\partial V}{\partial x}-\frac{\partial U}{\partial x}\right)+\frac{\partial U}{\partial y}\left(\frac{\partial V}{\partial y}-\frac{\partial U}{\partial y}\right)\right]d\Omega=0$$

За визначенням на границі Γ $U|_\Gamma=V|_\Gamma=U_0$ і маємо остаточний результат

$$\int_\Omega\left[\frac{\partial U}{\partial x}\left(\frac{\partial V}{\partial x}-\frac{\partial U}{\partial x}\right)+\frac{\partial U}{\partial y}\left(\frac{\partial V}{\partial y}-\frac{\partial U}{\partial y}\right)\right]d\Omega=0 \qquad (7.5.2)$$

Одержане співвідношення називається слабким формулюванням задачі Діріхле. Ця назва пояснюється тим, що лінійні функції є тривіальним для рішення Лапласа, а у отриманому формулюванні не тривіальними.

Екстремальне формулювання задачі Діріхле. Функція $f(z)$ називається опуклою, якщо для будь - яких $z, z_0 \in [a,b]$ справедлива умова

$$f(z_0) + f'(z_9)(z - z_0) \le f(z)$$

Нехай $$f(z) = z^2, z = \frac{\partial V}{\partial \partial z}, z_{0=} = \frac{\partial U}{\partial \partial z}$$

Тоді

$$f'(z_0) = 2\frac{\partial U}{\partial x}$$

і умова опуклості приймає вид

$$\left(\frac{\partial U}{\partial x}\right)^2 + 2\frac{\partial U}{\partial x}\left(\frac{\partial V}{\partial x} - \frac{\partial U}{\partial x}\right) \le \left(\frac{\partial V}{\partial x}\right)^2$$

Або

$$\frac{\partial U}{\partial x}\left(\frac{\partial V}{\partial x} - \frac{\partial U}{\partial x}\right) \le \frac{1}{2}\left[\left(\frac{\partial V}{\partial x}\right)^2 - \left(\frac{\partial U}{\partial x}\right)^2\right] \qquad (7.5.3)$$

Заміна змінної x на y приводить до аналогічного результату

$$\frac{\partial U}{\partial y}\left(\frac{\partial V}{\partial y} - \frac{\partial U}{\partial y}\right) \le \frac{1}{2}\left[\left(\frac{\partial V}{\partial y}\right)^2 - \left(\frac{\partial U}{\partial y}\right)^2\right] \qquad (7.5.4)$$

Якщо скласти ліві і праві частини (7.5.3) і (7.5.4), після інтегрування по області Ω з врахуванням слабого формулювання задачі, одержимо оцінку

$$\int_\Omega \left[\left(\frac{\partial U}{\partial x}\right)^2 + \left(\frac{\partial U}{\partial y}\right)^2\right] d\Omega \le \int_\Omega \left[\left(\frac{\partial V}{\partial x}\right)^2 + \left(\frac{\partial V}{\partial y}\right)^2\right] \qquad (7.5.5)$$

або

$$I(U) \le I(V).$$

Із (7.5.5) витікає, що із всіх можливих допустимих рішень дійсним є те, що доставляє мінімум функціоналу I. Одержане співвідношення називається екстремальним формулюванням задачі Діріхле. Постановка задачі у формі

(7.5.5) і (7.5.6) мають назву узагальнених постановок, а рішення, що їм відповідають, узагальненими.

Метод Рітца. розв'язку задач математичної фізики у варіаційній постановці є досить ефективними і розповсюдженими у обчислювальній практиці, а його ідея лягла в основу цілої низки інших методів.

Ідея метода Рітца полягає в тому, що допустиме рішення шукається у вигляді

$$V(x,y)=\varphi_0(x,y)+\sum_{k=1}^{n}c_k\varphi_k(x,y)$$

Тут φ_0 деяка функція, така що

$$\varphi_0(x,y)\big|_\Gamma=U_0(x,y)\ \text{, а}$$
$$\varphi_0(x,y)\big|_\Gamma=0\ ,k=1,2,\ldots,n$$

Таким чином задача зводиться до мінімізації функціоналу квадратичного відносно невідомих параметрів c_k. Після застосування необхідної умови мінімуму, задача зводиться до розв'язку СЛАР

$$\frac{\partial I}{\partial c_k}=0\,,\ k=1,2,\ldots,n \qquad (7.5.6)$$

Але при практичній реалізації цього досить ефективного і простого метода виникають певні труднощі. По – перше, нелегко підібрати функції φ_0 і φ_k, що задовольняють постановленим умовам. По – друге, для визначення коефіцієнтів СЛАР з (7.5.6) необхідно застосовувати чисельне інтегрування, сильно осцилюючих функцій, що приводить до значної обчислювальної похибки. І на кінець, матриця СЛАР є повністю заповненою і чутливої до похибок округлення. Далі розглядається метод , що є розвиненням метода Рітца і у якому приведені труднощі вдалось перебороти.

7.6 Метод скінчених елементів.

Цей метод (далі МСЕ), за більш чим півсторіччя свого існування, дякуючи своєї ефективності став самим розповсюдженим для рішення задач цього класу.

Ідея метода полягає в тому, що область Ω зображується у вигляді множини підобластей, у кожній з яких рішення дається у простій формі. Так на рис.7.6 зображена довільна область після дискретизації.

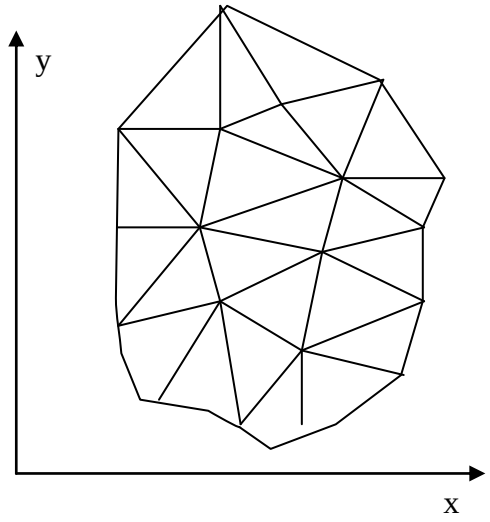


Рис 7.6.-Дискретизація скінченним елементом

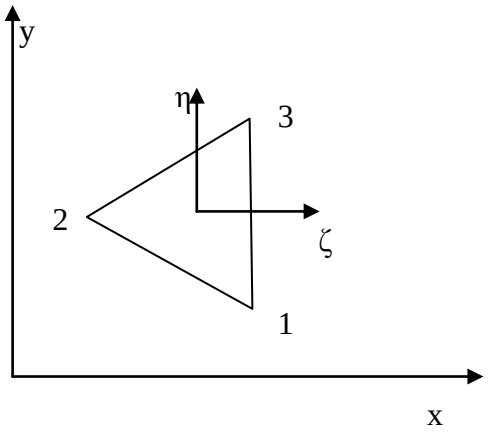


Рис 7.7 Локальна система координат в методі скінченних елементів

Такі області називаються скінченими елементами (СЕ) і можуть мати довільну форму, але при цьому повинна бути виконана умова суцільності, тобто не повинно бути розривів на границях СЕ. Найбільш поширену форму СЕ є підобласті з прямолінійними границями і частинами всього трикутника.

Поряд з глобальною системою координат $X0Y$ в середині кожного СЕ вводиться локальна система $\eta 0\varsigma$ (рис 7.7). В границях СЕ допустиме рішення приближається лінійною функцією локальних координат

$$V = a_0 + a_1\varsigma + a_2\eta$$

V – функція форми СЕ, параметри якої a_0, a_1, a_2 обчислюються однозначно по значенням функції в кутах V_1, V_2, V_3 з системи рівнянь

$$\begin{cases} a_{0+} + a_1\varsigma_1 + a_2\eta_1 = V_1 \\ a_{0+} + a_1\varsigma_2 + a_2\eta_2 = V_2 \\ a_{0+} + a_1\varsigma_3 + a_2\eta_3 = V_3 \end{cases}$$

Визначник цієї системи рівний

$$\Delta = \begin{vmatrix} 1 & \varsigma_1 & \eta_1 \\ 1 & \varsigma_2 & \eta_2 \\ 1 & \varsigma_3 & \eta_3 \end{vmatrix} = \begin{vmatrix} 1 & \varsigma_1 & \eta_1 \\ 0 & \varsigma_2 - \varsigma_1 & \eta_2 - \eta_1 \\ 0 & \varsigma_3 - \varsigma_1 & \eta_3 - \eta_1 \end{vmatrix}$$

Тобто Δ рівний подвійній площі СЕ і може бути рівним нулю тільки у тому випадку коли елемент стягується у точку. Тоді, наприклад для параметра a_0 маємо

$$a_0 = \begin{vmatrix} V_1 & \varsigma_1 & \eta_1 \\ V_2 & \varsigma_2 & \eta_2 \\ V_3 & \varsigma_3 & \eta_3 \end{vmatrix}$$

Звідки витікає, що параметри a_0, a_1, a_2 є лінійними функціями значень допустимої функції у кутах СЕ.

Так як функція форми є лінійною всередині СЕ, то вона є лінійною і на його границях. А якщо дві лінійні функції приймають рівні значення на кінцях відрізка, то вони рівні на всьому відрізку. Таким чином функції форми неперервні на границях СЕ. Очевидно, що похідні функції і форми є кусково постійними всередині області Ω

$$\frac{\partial V}{\partial x} = \frac{\partial V}{\partial \varsigma} = a_1^{(i)}$$

$$\frac{\partial V}{\partial y} = \frac{\partial V}{\partial \eta} = a_2^{(i)}$$

де i – номер елемента і вихідна задача зводиться до мінімізації квадратичної функції

$$\int_{\Omega}\left[\left(\frac{\partial V}{\partial x}\right)^2+\left(\frac{\partial V}{\partial y}\right)^2\right]d\Omega=\sum_{i=1}^{N}\int_{\Omega}\left[\left(a_1^{(i)}\right)^2+\left(a_1^{(i)}\right)^2\right]d\Omega \qquad (7.6.1)$$

Так як функція форми визначається тільки своїми значеннями в кутах СЕ, то вона відмінна від нуля тільки в тих елементах для яких деякий вузол є спільним (рис 7.8)

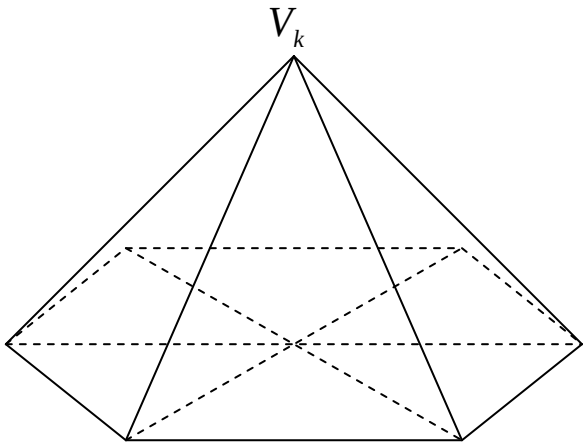


Рис 7.8- Графік функції з компактним носієм

Тобто таке наближення є локальним. До переваг скінчено – елементної

Апроксимації рішення слідує віднести:

- так як приближення мають локальний характер, легко задовольнити граничними умовами;

- операцію інтегрування по СЕ зводиться до простого перемноження константи на площу елемента;

матриця СЛАР яка знаходиться з необхідної умови мінімуму (7.6.1) діагонально – орієнтованою.

В даний час існує досить багато програмних пакетів, у яких реалізовано МСЕ. Вони забезпечують автоматичне розбиття області на елементи, а також мають зручні засоби завдання граничних умов. Приймаючи до уваги потужні обчислювальні ресурси сучасних ЕОМ, можна сказати, що цей метод є одним з найефективніших засобів рішення задач математичної фізики.

7.7 ЛАБОРАТОРНА РОБОТА № 8 Звичайні диференціальні рівняння

Завдання до лабораторної роботи. Знайти розв'язок задачі Коші для рівняння другого порядку

$$y'' = f(f, y, y'); \quad y(x_0) = y_0, \ y'(x_0) = y'_0$$

на вказаному інтервалі методом Рунне–Кутта четвертого порядку з автоматичним вибором кроку інтегрування.

Методичні вказівки до виконання лабораторної роботи.

Приведемо задачу Коші для рівняння другого порядку до системи рівнянь першого порядку

$$\begin{cases} z' = f(x, y, z), \\ y' = z = p(z); \end{cases} \quad y(x_0) = y_0; \ z(x_0) = z_0.$$

Згідно методу Рунне –Кутти розв'язок на $(n+1)$-му кроці визначається як

$$z_{n+1} = z_n + \Delta z, \qquad y_{n+1} = y_n + \Delta y,$$

де величини z_n, y_n є відомими, а $\Delta z, \Delta y$ обчислюються як

$$\Delta z = (k_1 + 2k_2 + 2k_3 + k_4)/6, \quad \Delta y = (l_1 + 2l_2 + 2l_3 + l_4)/6, \quad (4.7)$$

а допоміжні функції $k_i(h)$, $l_i(h)$, $i = 1,2,3,4$ визначаються наступним чином

$$k_1 = hf(x_n, y_n, z_n), \quad k_2 = hf(x_n + \frac{h}{2}, y_n + \frac{k_1}{2}, z_n + \frac{k_1}{2}),$$

$$k_3 = hf(x_n + \frac{h}{2}, y_n + \frac{k_2}{2}, z_n + \frac{k_2}{2}), \quad k_4 = hf(x_n + h, y_n + k_3, z_n + k_3).$$

$$l_1 = hp(x_n, y_n, z_n), \quad l_2 = hp(x_n + \frac{h}{2}, y_n + \frac{l_1}{2}, z_n + \frac{l_1}{2}),$$

$$l_3 = hp(x_n + \frac{h}{2}, y_n + \frac{l_2}{2}, z_n + \frac{l_2}{2}), \quad l_4 = hp(x_n + h, y_n + l_3, z_n + l_3).$$

Оскільки друге рівняння системи має спеціальний вид $p(z)=z$, то формули для обчислення Δy можна суттєво спростити, а саме

$$l_1 = hz_n,\ l_2 = hp(z_n + l_1/2),\ l_3 = h(z_n + l_2/2),\ l_4 = h(, z_n + l_3).$$

Для обчислення величин $\Delta z,\ \Delta y$ зручно використати дві процедури з заголовками виду

Procedure Runge_z(x,y,z,h:Real; Var dz:Real); (повертає величину Δz),

Procedure Runge_y(x,y,z,h:Real; Var dz:Real); (повертає величину Δy).

При цьому з процедури Runge_z викликається функція $f(x,y,z)$, а у процедурі Runge_y величина Δy обчислюється безпосереднє по формулам (4.7). Величина кроку інтегрування спочатку вибирається як $l_1 = \dfrac{b-a}{n}$, де a і b – кінцеві точки інтервалу, число вузлів інтегрування(нумерація ведеться з нуля).

Для оцінки похибки застосуємо правило Рунге і робоча формула набуває вигляду

$$\delta = \frac{32}{31}\left| y(x_n + h) - y\left(x_n + \frac{h}{2} + \frac{h}{2}\right)\right|.$$

Запис $y\left(x_n + \dfrac{h}{2} + \dfrac{h}{2}\right)$ означає, що обчислення функції у точці $x = x_n + h$ проводиться за два кроки. Позначимо

$$y_1 = y(x_n + h),\ y_2 = y\left(x_n + \frac{h}{2}\right) \text{ і } y_3 = y\left(x_n + \frac{h}{2} + \frac{h}{2}\right).$$

Тоді ці величини можна отримати як результат наступних викликів

Runge_y(x,y,z,h,dy);	y1:=y+dy;
Runge_z(x,y,z,h,dy);	z1:=z+dz;
Runge_y(x,y,z,h,dy);	y2:=y+dy;
Runge_z(x,y,z,h,dy);	z2:=y+dz;
Runge_y(x+0.5*h,y2,z2,h,dy);	y3:=y2+dy;

Runge_z(x+0.5*h,y2,z2,h,dy); z3:=z2+dz;

Контрольні питання.

Чим визначається порядок методів Рунне?

Від чого залежить точність розв'язку ЗДР?

Який з методів є більш точним: Ейлера, Рунге, апроксимації скінченими різницями і чому?

В яких з вище наведених методів є суттєвим обмеження на лінійність рівняння?

Який з вище наведених методів потребує найменшого обсягу обчислень?

Для яких рішень метод Ейлера є точним?

Варіанти завдань до лабораторної роботи.

Варіант	Рівняння	Початкові умови і інтервал інтегрування
1	$y''+\sin xy'-3x^2y=0$	$y(0)=0,\quad y'(0)=1,\quad [0;1]$
2	$y''+yy'+xy=0$	$y(1)=1,\quad y'(1)=1,\quad [1;2]$
3	$y''+\cos^2 xy'-y=0$	$y(0)=1,\quad y'(0)=0,\quad [0;1]$
4	$y''+\ln(x+2)y'+y=0$	$y(2)=0,\quad y'(2)=1,\quad [2;3]$
5	$y''+(x+y)^2y'-x^3y=0$	$y(0)=2,\quad y'(0)=0,\quad [0;1]$
6	$y''+\cos xy'+2x^2y=0$	$y(0)=1,\quad y'(0)=1,\quad [0;1]$
7	$y''+\sin(x+y)y'+y=0$	$y(-1)=0,\quad y'(-1)=0,\quad [-1;0]$
8	$y''+\sin(x^2+1)y'-xy=0$	$y(1)=0,\quad y'(1)=2,\quad [1;2]$
9	$y''+\cos(x+y)y'-\sin 3x^2=0$	$y(1)=1,\quad y'(1)=1,\quad [1;2]$
10	$y''+\sin xy'-e^xy=0$	$y(0)=0,\quad y'(0)=1,\quad [0;1]$
11	$y''+e^xy'-2x^2y=0$	$y(0)=1,\quad y'(0)=0,\quad [0;1]$
12	$y''+\sin xy'+e^{x+y}=0$	$y(0)=0,\quad y'(0)=1,\quad [0;1]$
13	$y''+y'-xe^y=0$	$y(0)=0,\quad y'(0)=0,\quad [0;1]$
14	$y''+\cos(x+1)y'+x^2y=0$	$y(0)=1,\quad y'(0)=1,\quad [0;1]$

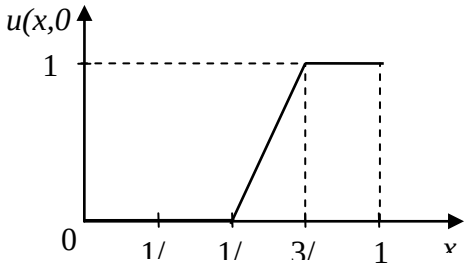
15	$y''+y'-\ln(x^2+1)y=0$	$y(-1)=1,\quad y'(-1)=1,\quad [-1;0]$
16	$y''+\sin(x-y)y'+y=0$	$y(0)=0,\quad y'(0)=1,\quad [0;1]$
17	$y''+\sin y+x^2y=0$	$y(2)=0,\quad y'(2)=1,\quad [2;3]$
18	$y''+\cos y+\sin xy=0$	$y(1)=1,\quad y'(1)=0,\quad [1;2]$
19	$y''+\cos yy'+\sin(xy\}=0$	$y(-1)=0,\quad y'(-1)=1,\quad [-1;1]$
20	$y''+xe^y+x=0$	$y(0)=0,\quad y'(0)=1,\quad [0;1]$
21	$y''+\cos yy'-xy=0$	$y(0)=0,\quad y'(0)=1,\quad [0;2]$
22	$y''+y'-y^3=0$	$y(\pi)=0,\quad y'(\pi)=1,\quad [\pi;1.5\pi]$
23	$y''+x^2y'+y^2=0$	$y(0)=1,\quad y'(0)=1,\quad [0;1]$
24	$y''+\cos(x^2+x+1)y'+xy=0$	$y(0)=1,\quad y'(0)=0,\quad [0;1]$
25	$y''-3x^2y'-\sin xy=0$	$y(-1)=0,\quad y'(-1)=0,\quad [-1;1]$

7.8 ЛАБОРАТОРНА РОБОТА № 9 **Рівняння типу теплопровідності**

Завдання до лабораторної роботи. Отримати розподіл температури по довжині стержня з часом за допомогою явної і неявної скінченно-різницевих схем.

Методичні вказівки до виконання лабораторної роботи.

Розглянемо задачу теплопровідності в стержні, початковий розподіл температури в якому має вигляд



На лівому кінці підтримується температура $u|_{x=0}=0$, а правий кінець теплоізольований, тобто $\left.\frac{\partial u}{\partial x}\right|_{x=1}=0$. Температуру будемо зберігати у вигляді двовимірного масиву дійсних чисел

U: array [0..n,0..100] of real;

де n – кількість вузлів. Перший індекс масиву відповідає координаті, другий – часу.

Після обчислення кроку по координаті

h:=1/(n+1)

задаємо величину кроку по часу dt. При проведенні обчислень по явній скінчено різницевій схемі для забезпечення стійкості схеми треба дотримуватись умови $r=\frac{dt}{h^2}\le\frac{1}{2}$, тобто

r:=1/4; *//як приклад*

dt:=r*h^2.

Відповідно до графіку розподілу температури в момент $t=0$ задаємо початкові умови. Рівняння прямої на відрізку $[1/2;3/4]$ виведемо, скориставшись рівнянням прямої, що проходить через дві точки з відомими координатами (x_1,y_1) і (x_2,y_2):

$$\frac{x-x_1}{x_2-x_1}=\frac{y-y_1}{y_2-y_1}.$$

Для заданих початкових умов $x_1=1/2$, $y_1=0$, $x_2=3/4$, $y_2=1$, і рівняння прямої набуває вигляду

$$\frac{x-1/2}{3/4-1/2}=\frac{y-0}{1-0}.$$

Після відповідних перетворень отримуємо

$$y=4x-2.$$

Тоді завдання початкових умов буде мати вигляд

```
x:=0;
For i:=0 to n do
    Begin
        U[i,0]:=0;
        If (x>=1/2) and (x<3/4) then U[i,0]:=4*x-2;
        If (x>=3/4) and (x<=1) then U[i,0]:=1;
        x:=x+h;
    End.
```

Далі шар за шаром обчислюємо температуру по явній скінченно-різницевій схемі, паралельно організовуючи вивід:

```
j:=0;
t:=0;
Do
    u[0,j+1]:=0;                                    (*)
    For i:=1 to n-1 do
        u[i,j+1]:=u[i,j]+r*(u[i-1,j]-2*u[i,j]+u[i+1,j]);
    u[n,j+1]:=u[n-1,j+1];                           (**)
    t:=t+dt;
    write(t,` `);  for i:=0 to n do write(u[i,j],` `); writeln;
    j:=j+1;
Until t<tk;
```

де tk – кінцевий момент часу.

Рядки (*) і (**) забезпечують виконання граничних умов, оскільки $u\big|_{x=0} = u[0, j+1] = 0$ і $\left.\frac{\partial u}{\partial x}\right|_{x=1} = \frac{u[n, j+1] - u[n-1, j+1]}{h} = 0$.

Обчислення по неявній скінченно-різницевій схемі проводяться за допомогою методу прогонки для розв’язання системи лінійних алгебраїчних рівнянь з трьох-діагональною структурою. Перед викликом відповідної процедури, треба задавати коефіцієнти цієї системи.

Контрольні питання.

У чому полягає задача Коші?

Чим визначається порядок методів Рунге?

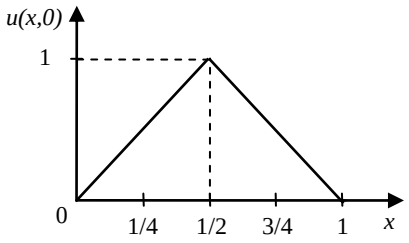
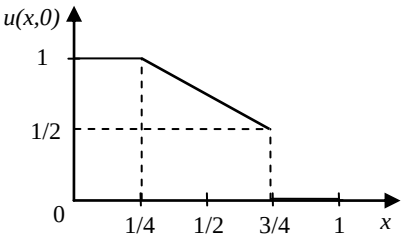
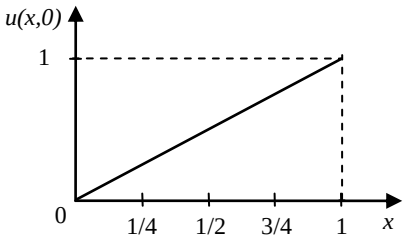
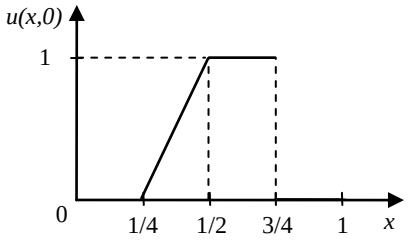
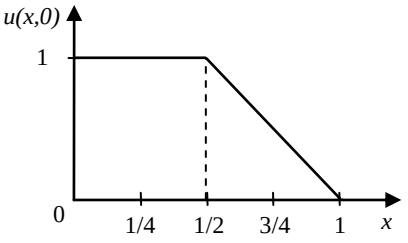
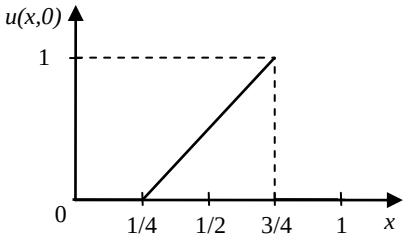
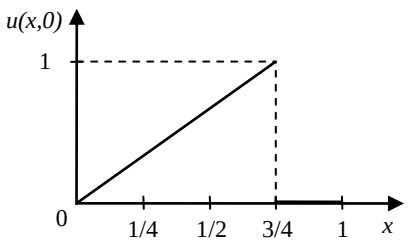
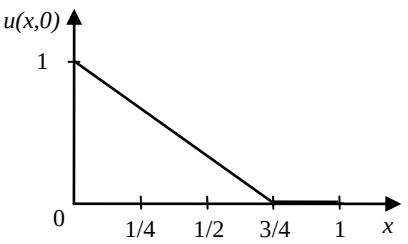
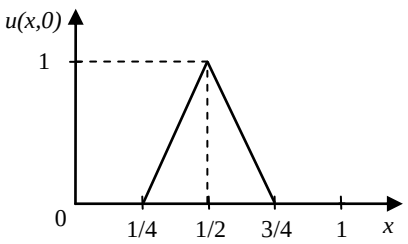
Від чого залежить точність розв'язку диференціального рівняння в часткових похідних за допомогою скінчено- різницевої схеми?

Яка зі схем (явна чи неявна) є стійкою за яких умов?

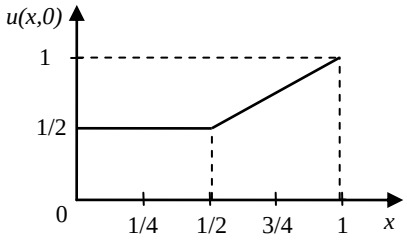
Яка зі схем дає більш точний результат, явна чи неявна, і чому?

Яка схема (явна чи неявна) потребує найменшого обсягу обчислень?

Варіанти завдань.

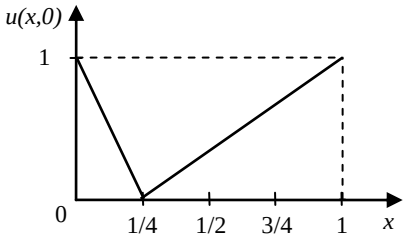
1.  $u\|_{x=0}=0,\quad \left.\frac{\partial u}{\partial x}\right\|_{x=1}=0.$	2.  $u\|_{x=0}=1,\quad \left.\frac{\partial u}{\partial x}\right\|_{x=1}=1.$	3.  $\left.\frac{\partial u}{\partial x}\right\|_{x=0}=0,\quad u\|_{x=1}=1.$
4.  $\left.\frac{\partial u}{\partial x}\right\|_{x=0}=0,\quad u\|_{x=1}=0.$	5.  $u\|_{x=0}=1,\quad \left.\frac{\partial u}{\partial x}\right\|_{x=1}=0.$	6.  $u\|_{x=0}=0,\quad \left.\frac{\partial u}{\partial x}\right\|_{x=1}=1.$
7.  $\left.\frac{\partial u}{\partial x}\right\|_{x=0}=1,\quad u\|_{x=1}=0.$	8.  $u\|_{x=0}=1,\quad u\|_{x=1}=0.$	9.  $\left.\frac{\partial u}{\partial x}\right\|_{x=0}=0,\quad \left.\frac{\partial u}{\partial x}\right\|_{x=1}=0.$

10.



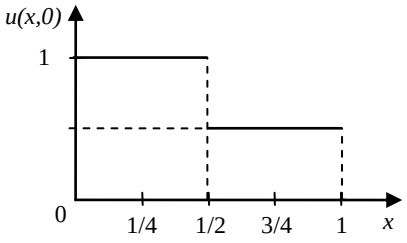
$u\big|_{x=0} = \frac{1}{2}, \qquad \frac{\partial u}{\partial x}\Big|_{x=1} = 0.$

11.



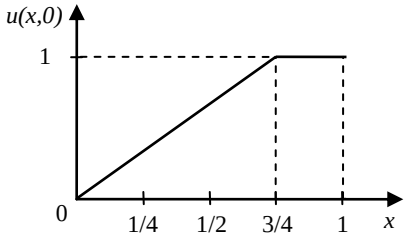
$u\big|_{x=0} = 1, \qquad u\big|_{x=1} = 1.$

12.



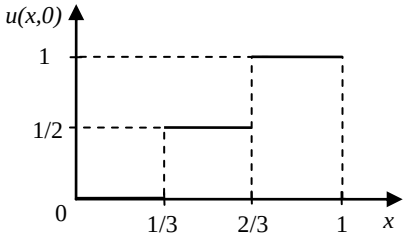
$\frac{\partial u}{\partial x}\Big|_{x=0} = 0, \qquad \frac{\partial u}{\partial x}\Big|_{x=1} = 1.$

13.



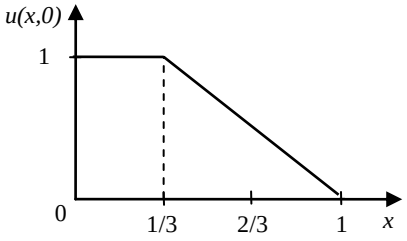
$u\big|_{x=0} = 0, \qquad \frac{\partial u}{\partial x}\Big|_{x=1} = 0.$

14.



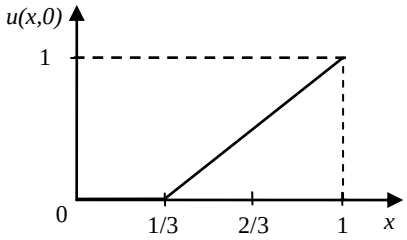
$\frac{\partial u}{\partial x}\Big|_{x=0} = 0, \qquad u\big|_{x=1} = 1.$

15.



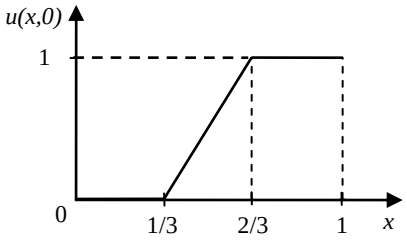
$u\big|_{x=0} = 1, \qquad \frac{\partial u}{\partial x}\Big|_{x=1} = 0.$

16.



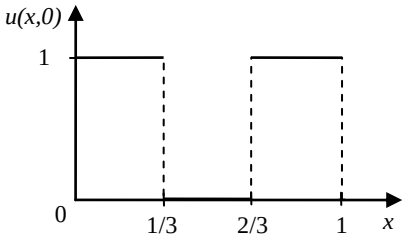
$\frac{\partial u}{\partial x}\Big|_{x=0} = 1, \qquad u\big|_{x=1} = 1.$

17.



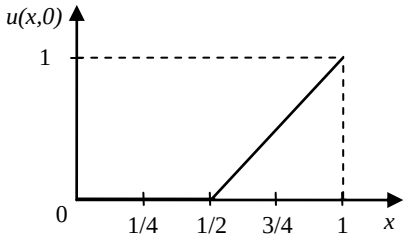
$u\big|_{x=0} = 0, \qquad \frac{\partial u}{\partial x}\Big|_{x=1} = 0.$

18.



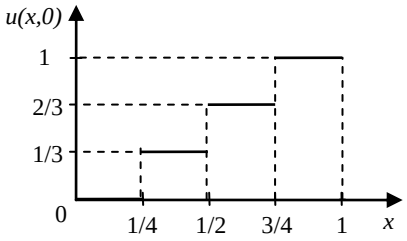
$\frac{\partial u}{\partial x}\Big|_{x=0} = 0, \qquad \frac{\partial u}{\partial x}\Big|_{x=1} = 0.$

19.



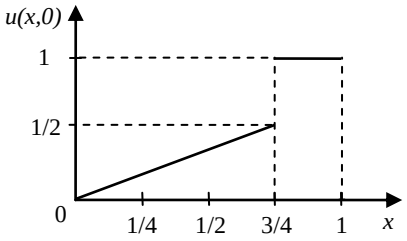
$\frac{\partial u}{\partial x}\Big|_{x=0} = 0, \qquad u\big|_{x=1} = 1.$

20.



$\frac{\partial u}{\partial x}\Big|_{x=0} = 1, \qquad \frac{\partial u}{\partial x}\Big|_{x=1} = 0.$

21.

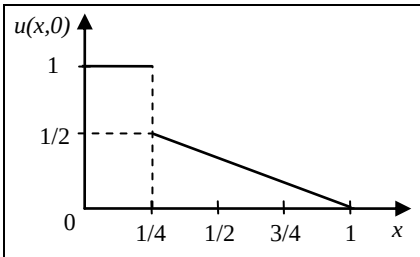
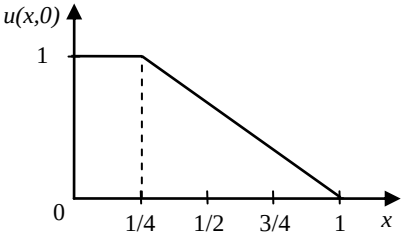
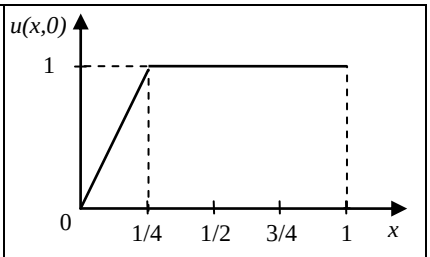
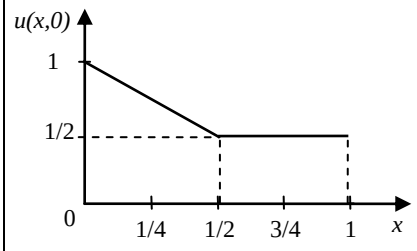


$\frac{\partial u}{\partial x}\Big|_{x=0} = 1, \qquad u\big|_{x=1} = 1.$

22.

23.

24.

 $u\big\|_{x=0}=1,\qquad u\big\|_{x=1}=0.$	 $\dfrac{\partial u}{\partial x}\Big\|_{x=0}=0,\qquad u\big\|_{x=1}=0.$	 $u\big\|_{x=0}=0,\qquad u\big\|_{x=1}=1.$
25.  $u\big\|_{x=0}=1,\qquad \dfrac{\partial u}{\partial x}\Big\|_{x=1}=0.$		

РОЗДІЛ 8
МЕТОДИ ОПТИМІЗАЦІЇ

До задач на пошук оптимуму зводяться багато проблем математики, системного аналізу, техніки, економіки, медицини. Зокрема вони виникають при побудові математичних моделей , коли для вивчення якогось складного явища розробляється модель і до оптимізації вдаються, щоб визначити таку структуру і такі параметри останньої, які забезпечують найкращу відповідність останньої до реальності. Іншою традиційною областю використання

оптимізації є процедури прийняття рішень, так як більшість з них націлена саме на те, щоб зробити оптимальній вибір.

Задача оптимізації існує в одному з двох наступних формулювань

$$\textit{знайти}\ \min\ f(x)\ \ x \in R^n \ \text{або}\ \ \textit{знайти}\ \max\ f(x)\ \ x \in R^n,$$

де R - поле дійсних чисел. Така задача називається задачею без обмежень або задачею безумовної оптимізації .

Якщо крім розв'язку (8.3.1) необхідно, щоб рішення задовольняло умовам

$$g_i(x) = 0; \quad \text{i} = 1,2,...,\text{m}$$

$$h_j(x) \le 0; \quad \text{j} = 1,2,...,\text{k}, m, k \le n,$$

то маємо задачу з обмеженнями або задачу умовної оптимізації .

При $n = 1$ маємо задачу одновимірної оптимізації, при $n > 1$ коли $f(x) = f(x_1, x_2, ..., x_n)$ - задачу багатовимірної оптимізації.

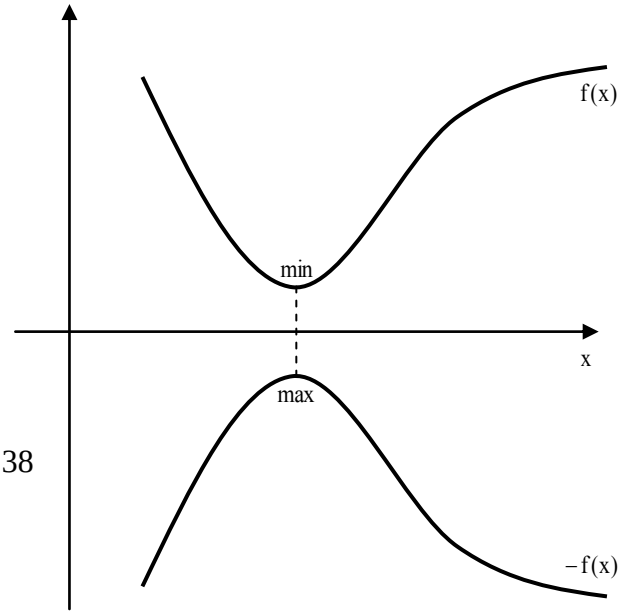
Дамо визначення локального мінімуму. Якщо для деякого числа $\delta > 0$ виконується умова

$$f(x^*) \le f(x) \qquad (8.1)$$

при $\left|x - x^*\right| \le \delta,$ то функція у точці x^* має *локальний мінімум*. Якщо ж умова (8.1) виконується для будь якого δ, то цьому разі мінімум є глобальним. Точка x^* називається оптимальною а функція $f(x)$ цільової.

8.1 Одновимірна оптимізація

Всі існуючі методи рішення цієї задачі розраховані тільки на випадок локального мінімуму. Звичайно задача оптимізації формулюється як пошук мінімуму. У тому випадку коли необхідно знайти максимум достатньо



поміняти знак цільової функції на протилежний.

Очевидно, що одномірна оптимізація є частковим випадком багатовимірної, але більшість методів рішення багатовимірної задачі включають методи для знаходження екстремуму функції одної змінної.

Рис.8.1.- Зв'язок задач на мінімум і максимум

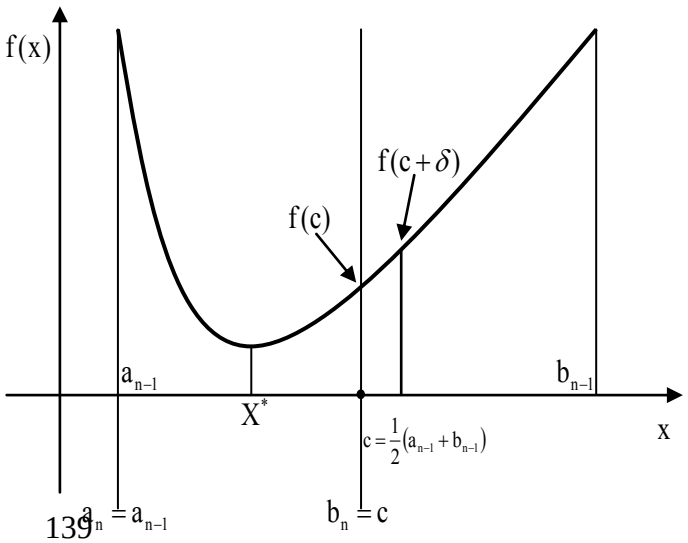
Методи ***рішення одновимірної оптимізації.*** У тому разі коли цільова функцію є диференційованою, то в силу необхідної умови існування екстремуму задача *знайти* $\min f(x)$ приводиться до розв'язку нелінійного рівняння $\varphi(x) = f'(x) = 0,$ рішення якого можна знайти використовуючи методи розділу 5. У протилежному випадку застосовують наступні методи.

Метод дихотомії, як і метод половинного ділення побудований на послідовному зменшенні довжини інтервалу невизначеності. Але у цьому випадку інтервал містить не корінь рівняння, а оптимальну точку. На кожній ітерації обчислюється значення цільової функції у двох точках $f(c)$ і $f(c+\delta),$ де $c = \frac{1}{2}(a+b)$. Тут δ деяке мале число, величина якого вибирається з умови, щоб значення $f(x)$ і $f(x+\delta)$ можна було розрізнити. Після цього виходячи з співвідношень

$$f(c+\delta) > f(c) \Rightarrow a_{k+1} = a_k, b_{k+1} = c \qquad (8.1.1)$$

$$f(c+\delta) \le f(c) \Rightarrow a_{k+1} = c, b_{k+1} = b_k$$

формується новий інтервал і перевіряється виконання умови точності ($|a_k - b_k| \le \varepsilon$). Так ситуація, що приведена на рис. 8.2 відповідає першому співвідношенню у (8.1.1). Суттєвим недоліком методу дихотомії є те, що на кожній



ітерації значення цільової функції доводиться обчислювати двічі.

Рис 8.2.-Схема метода дихотомії

За міру ефективності методу звичайно приймається кількість обчислень цільової функції, яку він потребує. Цілком природно спробувати побудувати метод, у якому цільова функція на ітерації обчислювалась б тільки один раз.

Точка c є точкою золотого перерізу для відрізка довжиною l, якщо вона ділить його на дві частини з довжинами l_1 і l_2 відповідно у співвідношенні

$$\frac{l_1}{l} = \frac{l_2}{l_1}$$

Оскільки $l = l_1 + l_2$ то довжини цих відрізків можна визначити з квадратного рівняння $l_1^2 = l^2 - ll_1$, звідки $l_1 = \frac{l}{2}(\sqrt{5}-1)$, $l_2 = \frac{l}{2}(3-\sqrt{5})$.

Легко бачити, що для кожного відрізка ab існують дві точки золотого перерізу ліва - α і права - β, при цьому

$$ab = l,\ a\alpha = \beta b = l_2 \quad a\beta = \alpha b = l_1.$$

Якщо у ab відкинути праву частину βb, то ліва точка α початкового відрізка буде правою для $a\beta$, і навпаки. Дійсно для відрізка $a\beta$ маємо

$$\frac{a\alpha}{a\beta} = \frac{l_2}{l_1} = \frac{3-\sqrt{5}}{\sqrt{5}-1} = \frac{1}{2}(\sqrt{5}-1)$$

а для відрізка αb $\frac{\alpha\beta}{\alpha b} = \frac{l(\sqrt{5}-2)}{l_1} = \frac{1}{2}(3-\sqrt{5})$. Ця властивість і є ключем до ***методу золотого перерізу.*** До початку роботи метода необхідно обчислити два значення цільової функції $f(\alpha_k)$, $f(\beta_k)$, (k - номер ітерації), а наступний інтервал $a_{k+1}b_{k+1}$ невизначеності формується, виходячи з наступних двох ситуацій

$$f(\alpha_k) < f(\beta_k) \ \Rightarrow\ \beta_{k+1} = \alpha_k,\ a_{k+1} = \alpha_k,\ b_{k+1} = \beta_k,\ f(\beta_{k+1}) = f(\alpha_k)$$

$$f(\beta_k) < f(\alpha_k) \ \Rightarrow\ \alpha_{k+1} = \beta_k,\ a_{k+1} = \alpha_k,\ b_{k+1} = b_k,\ f(\alpha_{k+1}) = f(\beta_k)$$

і таким чином треба знайти тільки одне значення цільової функції, постільки в другій точці золотого перерізу воно вже відоме.

Методі Фібоначчі. Особливістю цього метода є те що оптимальна точка повинна бути знайдена за задане число ітерацій. Нехай наступний інтервал невизначеності l_n зв'язаний з попереднім l_{n-1} як $l_n = \frac{l_{n-1} + \varepsilon}{2}$ (рис. 8.3а) або

$$l_{n-1} = 2l_n - \varepsilon \qquad (8.1.2)$$

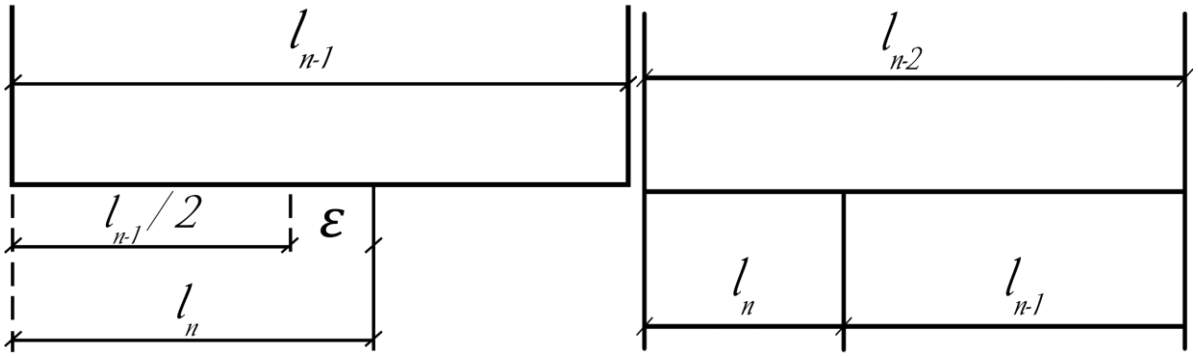


Рис. 8.3а Рис. 8.3б

З іншого боку довжини трьох послідовних інтервалів $l_{n-2} = l_n + l_{n-1}$ (рис. 8.3б), а для попереднього крок у

$$l_{n-3} = l_{n-1} + l_{n-2} = 2l_{n-1} + l_n \qquad (8.1.3)$$

З врахуванням (8.1.2) маємо $l_{n-3} == 5l_n - 2\varepsilon$. Числові коефіцієнти в ( ) є відомими числами Фібоначчі, які визначаються за рекурентною формулою

$$\Phi_n = \Phi_{n-1} + \Phi_{n-2}, \;\; \Phi_0 = \Phi_1 = 1$$

За допомогою рекурентної формули обчислимо $\Phi_2 = 2$, $\Phi_3 = 3, \Phi_4 = 5$, що дозволяє записати співвідношення (8.1.3) у формі $l_{n-3} = \Phi_4 l_n - \Phi_2 \varepsilon$ або у більш загальному вигляді

$$l_{n-k} = \Phi_{k+1} l_n - \Phi_{k-1} \varepsilon \qquad (8.1.4)$$

Якщо оптимальна точка повинна бути знайдена за N ітерацій , то покладаючи в (8.1.4) $k = N - 1$ приходимо до співвідношення, яке пов'язує визначимо довжини стартового та кінцевого інтервалів

$$l_1 = \Phi_N l_N - \Phi_{N-2} \varepsilon \ (\ )$$

Покладаючи в (8.1.4) $k = N - 2$, після виключення l_N за допомогою останнього співвідношення , маємо

$$l_2 = \Phi_{N-1} \left(\frac{L_1}{\Phi_N} - \frac{\Phi_{N-2}}{\Phi_N} \varepsilon \right) l_N - \Phi_{N-3} \varepsilon$$

З врахування малості ε отримаємо $l_2 = \frac{\Phi_{N-1}}{\Phi_N} l_1$.

Зберігаючи позначення , використані в методі золотого перетину , для лівої і правої точок нового інтервалу відповідно маємо

$$\alpha = b - l_2 = a + \frac{\Phi_{n-2}}{\Phi_n}(b - a)$$

$$\beta = a + l_2 = a + \frac{\Phi_{n-1}}{\Phi_n}(b - a)$$

Тоді для k-го кроку метода формування інтервалу здійснюється як

$$\alpha_k = a_k + \frac{\Phi_{n-k-1}}{\Phi_{n-k+1}}(b_k - a_k)$$

$$\beta_k = a_k + \frac{\Phi_{n-k}}{\Phi_{n-k+1}}(b_k - a_k)$$

При цьому коефіцієнт зменшення довжини інтервалу невизначеності дорівнює $\frac{\Phi_{n-k-1}}{\Phi_{n-k+1}}$, тобто не є сталою величиною. Також можна показати ,що .

$$\lim_{k\to\infty}\frac{\Phi_{k-1}}{\Phi_k}=\frac{2}{1+\sqrt{5}}=0{,}618,$$

тобто метод золотого перетину є граничним випадком методу Фібоначчі.

Всі розглянуті методи(дихотомії,золотого перетину половинного ділення і Фібоначчі) мають лінійну швидкість збіжності.

Метод параболічної апроксимації має в своїй основі наближення цільової функції загального виду квадратичною $f(x)\approx ax^2+bx+c$ і тоді x^* одразу знаходиться з умови

$$f'(x^*)==0,\ f'(x^*)=2ax^*+b=0$$

Звідки для оптимальної точки маємо $x^*=-\frac{b}{2a}$

Параметри a,b можуть бути визначені декілька ми способами, Наприклад з системи лінійних рівняння по відомим значенням цільової функції у трьох точках, з розкладення цільової функції в ряд Тейлора, але більш доцільним є наближення цільової функції квадратичним інтерполяційним поліномом у формі Ньютона, як це зроблено у розд.5 при розв'язку нелінійних рівнянь, формула (5.5.5).

8.2 Рішення задач багатовимірної оптимізації

За винятком квадратичної цільової функції, коли оптимальна точка знаходиться шляхом розв'язку СЛАР яка є достатньою умовою мінімуму, для рішення задач цього класу застосовуються ітераційні методи. Існує наступна класифікація цих методів:

- *методи нульового порядку* . Для пошуку мінімуму використовується інформація тільки про значення цільової функції.

- *методи першого порядку*. Для пошуку мінімуму використовується інформація як про значення цільової функції, так і про значення її перших похідних.
- *методи вищих порядків*. Для пошуку мінімуму використовується інформація як про значення цільової функції, так і значення її похідних вище першої.

Метод по координатного спуску побудований на зведенні задачі багатовимірної оптимізації до послідовності одновимірних задач. Якщо всі аргументи цільової функції окрім x_1 вважати фіксованим $x_i = x_i^{(0)}$, $\mathrm{i} = 2,..,\mathrm{n}$, то замість початкової багатовимірної задачі отримаємо одновимірну

$$\textit{знайти} \qquad \min \varphi_1^{(1)}(x_1), \qquad (8.2.1)$$

де $\varphi_1^{(1)}(x_1) = f(x_1, x_2^{(0)}, x_3^{(0)}, ..., x_n^{(0)})$

Нехай $x_1^{(1)}$ є оптимальна точка для задачі (8.2.1). Тоді наступний крок є задачею одновимірної оптимізації з цільовою функцією

$$\varphi_2^{(1)}(x_2) = f(x_1^{(1)}, x_2, x_3^{(0)}, ..., x_n^{(0)})$$

А цільова функція для уточнення i - ої компоненти на $(k+1)$ - ій ітерації набуде вигляду

$$\varphi_{\mathrm{i}}^{(\mathrm{k}+1)}(x_i) = f(x_1^{(k+1)}, ..., x_{i-1}^{(k+1)}, x_{i+1}^{(k)} ..., x_n^{(k)})$$

По формі цей метод співпадає з методом Зейделя розв'язку СЛАР і природно можна збільшити швидкість його збіжності, якщо перехід до наступної ітерації робити за формулою метода релаксації (5.2.10). Метод по координатного спуску є дуже ефективним коли по більшості змінних цільова функція є квадратичною. Тоді розв'язок одновимірної задач знаходиться безпосередньо з (8.1.3).

Розглянемо ще один з методів нульового порядку, який має кілька назв: ***метод деформованого багатогранника,*** сиплекс – метод, метод Нелдера-Міда.

$$x^{(n+1)} = x^{(n)} + ph$$

Цей метод базується на послідовності перетворень геометричного об'єкту, який називається симплексом. В просторі n вимірів симплекс є багатогранником, який визначається своїми $n+1$ вершинами . У випадку двох вимірів це трикутник, трьох – тетраедр. Існує декілька варіантів цього методу, в тому числі і узагальнення на випадок задачі умовної оптимізації.

Побудуємо багатогранник і обчислимо значення цільової функції в його $n+1$ вершині. Ці значення упорядковуються по зростанню, тобто $f_1 < f_2 < ... < f_{n+1}$. Таким чином найкращою точкою буде x_1, а найгіршою - x_{n+1}. Ідея метода полягає в тому, що на кожному кроці багатогранник перебудовується таким чином, що значення цільової функції в найгіршій точці зменшується, тобто покращується. По n кращим вершинам знайдемо середню точку $c = \frac{1}{n}\sum_{i=1}^{n} x_i$ і проведемо відносно неї відбиття найгіршої точки

$$x_r = c + \alpha(c - x_{n+1}).$$

Параметр $\alpha > 0$ регулює відстань між точкою x_{n+1} і її новим відбитим положенням x_r. Значенню $\alpha = 1$ відповідає випадок симетричного відбиття.

При цьому точка x_{n+1} замінюється на x_r тільки при виконуванні умови $f_r < f_n$. Якщо ця умова не виконується , тобто для відбитої точки маємо $f_n < f_r < f_{n+1}$ і в новому багатограннику відбита точка і в новому буде знову найгіршою , що приводить до за циклювання методу. У цьому випадку багатогранник стискується в сторону найкращої точки, як правило зменшенням ребер вдвічі $x_i = \frac{1}{2}(x_1 + x_i)$

Якщо в результаті відбиття $f_r < f_1$, робиться спроба розтягнути багатогранник

$$\tilde{x} = x_r + \beta(x_r - c) \quad \beta > 1,$$

де $\beta > 1$- коефіцієнт розтягування.

Ознакою закінчення роботи алгоритму є виконання умови

$$\|x_1 - x_2\| \le \varepsilon$$

Ітераційна схема більшості методів багатовимірної оптимізації має вигляд

$$x_{k+1} = x_k + hp_k,$$

де k - номер ітерації, p_k - вектор напрямок пошуку, h - довжина кроку у знайденому напрямку.

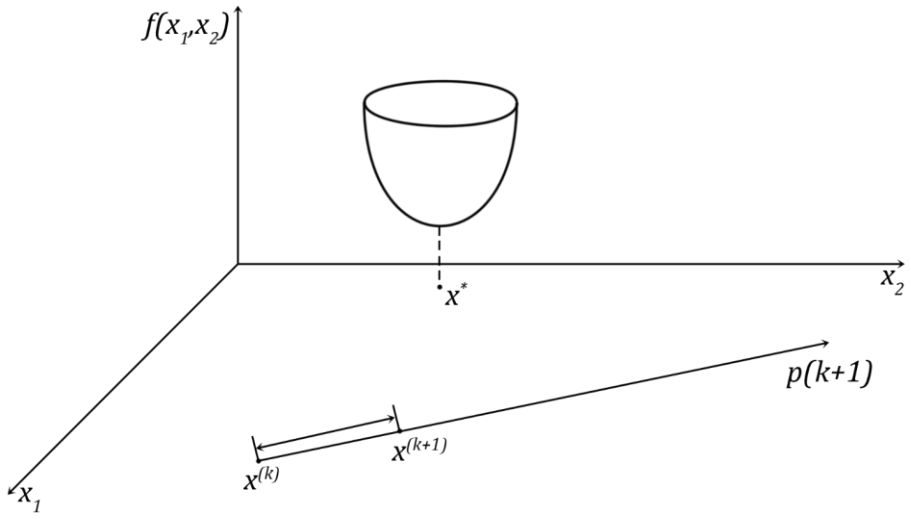


Рис 8.4. Схема пошук у випадку багатовимірної оптимізації

Тобто задача розбивається на дві під задачі: визначення напрямку p і одновимірний пошук у знайденому напрямку. Ефективність методу в основному визначається тим наскільки точно буде знайдено напрямок. Але бажано і навіть суттєво, щоб крок h_k був хорошим наближенням точки мінімуму функції f вздовж напрямку p_k.

По-перше це забезпечує «суттєве» зменшення f, а по-друге, висока швидкість збіжності багатьох методів спуску досягається тільки у тому випадку, коли крок вибирається саме цим способом. Практичним критерієм придатності кроку h_k як наближеного розв'язку одновимірної мінімізації полягає в вимозі, щоб модуль похідної функції f вздовж напрямку p_k у точці $x_k + h_k p_k$ був суттєво меншим, ніж у точці x_k:

$$\left| g(x_k + h_k p_k)^T p_k \right| \le -\eta g^T(x_k) p_k$$

Тут η число з діапазону $0 \le \eta \le 1$ що визначає точність, з якою h_k апроксимує стаціонарну точку f вздовж напрямку p_k. Шляхом варіації значення η, можна тим самим управляти трудомісткістю обчислень h_k. Обчислення h_k на основі критерію ( ) називають «акуратним одновимірним пошуком», а якщо $\eta = 0$, кажуть про «точний одновимірний пошук».

Якщо для рішення задачі використовуються тільки значення цільової функції, то такі методи називаються методами нульового порядку. В методах першого порядку окрім значення цільової функції використовується ще інформація про її перші похідні. Методи які ще потребують інформації про похідні вище першого прядку відносяться до методів вищих порядків.
Типовим представником методів першого порядку(вони ще називаються градієнтними) є метод най скорішого спуску. Тут за вектор напрямку пошуку вибирається вектор протилежний градієнту

$$p = -\frac{grad(f(x^{(k)}))}{\left| grad(f(x^{(k)})) \right|},$$

Далі ми розглянемо алгоритм який є історично першим з методів,побудованих на основі квадратичної апроксимації, що мінімізує функцію f. Така апроксимація, залишаючись достатньо простою, водночас

набагато точніша ніж лінійна, яка використовується в класичному методі найскорішого спуску. Це дозволяє на її основі будувати ефективні алгоритми.

Маючи перші і другі похідні цільової функції f , в якості її квадратичної моделі можна взяти суму перших трьох членів її розкладення в ряду Тейлора в околі поточної точки x_k , тобто скористуватися наближеною рівністю виду

$$f(x_k + p_k) \approx f_k + g_k^T p + \frac{1}{2} p^T G p$$

де вектор $g_i = \frac{\partial f}{\partial x_i}$ і матриця других похідних $G_{ij} = \frac{\partial^2 f}{\partial x_i \partial x_j}$ визначаються як

$$g = \begin{bmatrix} \frac{\partial f}{\partial x_1} \\ \frac{\partial f}{\partial x_2} \\ \vdots \\ \frac{\partial f}{\partial x_n} \end{bmatrix} \qquad G = \begin{bmatrix} \frac{\partial^2 f}{\partial x_1 \partial x_1} & \frac{\partial^2 f}{\partial x_1 \partial x_2} & \cdots & \frac{\partial^2 f}{\partial x_1 \partial x_n} \\ \frac{\partial^2 f}{\partial x_2 \partial x_1} & \frac{\partial^2 f}{\partial x_2 \partial x_2} & \cdots & \frac{\partial^2 f}{\partial x_2 \partial x_n} \\ \vdots & \vdots & \vdots & \vdots \\ \frac{\partial^2 f}{\partial x_n \partial x_1} & \frac{\partial^2 f}{\partial x_n \partial x_2} & \cdots & \frac{\partial^2 f}{\partial x_n \partial x_n} \end{bmatrix}$$

Мінімум її правої частини(якщо він існує) досягається на векторі p_k, що доставляє мінімум квадратичній формі

$$\Phi(p) \approx g_k^T p + \frac{1}{2} p^T G p$$

Будучи стаціонарною точкою, цей вектор задовольняє умові $\frac{\partial \Phi}{\partial p} = 0$, яка очевидним чином може бути записана як

$$G p_k = -g_k \qquad (8.2.2)$$

Тобто приводить до СЛАР для визначання вектора p .

Алгоритм мінімізації, в якому напрямок визначається системою ( ) називається ***методом Ньютона***, а розв'язок цієї системи – ***ньютоновським напрямком.***

Метод Ньютона, належить до методів другого порядку і є багатовимірним аналогом метода параболічної апроксимації.

В задачах мінімізації довільної квадратичної функції з додатно визначеною матрицею G, метод Ньютона дає розв'язок за одну ітерацію незалежно від вибору початкового наближення, при цьому крок h_k повинен бути одиничним. Достатньою умовою збіжності метода Ньютона є додатна визначеність матриці Гессе цільової функції.

Висока швидкість локальної збіжності методу Ньютона робить його надзвичайно привабливим для використання і він є своєрідним еталоном серед алгоритмів безумовної оптимізації. Гарантією ефективності методів ньютонівського типу є врахування інформації про кривизну цільової функції, яка міститься в матриці Гессе і дозволяє будувати локально точні квадратичні моделі.

В методі Ньютона на кожній ітерації необхідно обчислити елементи матриці других похідних G і розв'язати систему лінійних рівнянь. В залежності від функції f складність кожної ітерації може складати $O(n^3)$, що викликає труднощі при застосуванні метода Ньютона для задач середньої і великої розмірності. Теорія квазіньютоновських методів опирається на можливість апроксимації кривизни нелінійної функції без явного обчислення матриці Гессе. На кожній ітерації нове наближення матриці Гессе B_{k+1} будується так, щоб виконувалася умова, яка називається квазіньютоновською і має вид

$$B_{k+1}s_k = y_k \qquad (8.2.3)$$

Тут $s_k = x_{k+1} - x_k$, $y_k = g(x_{k+1}) - g(x_k)$

При такому підході щоб уникнути розв'язку системи лінійних рівнянь можна зразу будувати наближення не до матриці Гессе, а до оберненої гессіану матриці H_k. Матриці H_k будуються таким чином,щоб у граничному випадку забезпечити апроксимацію істинного оберненого гессіану

$$H_k - G^1 \to 0 \text{ при } k \to \infty$$

Якби матриця H_k точно дорівнювала оберненій матриці Гессе G^1, то б це був просто метод Ньютона. Звідси і походить назва квазіньютоновські методи. У цьому випадку квазіньютоновська умова набуває форми

$$H_{k+1}y_k = s_k \quad (8.2.4)$$

В одномірному випадку формули (8.2.3 .(8.2.4) є рівнянням січної, звідки і їх друга назва – умова січної. У випадку коли вимір задачі є більшим одиниці, співвідношення (8.2.3).(8.2.4) є недовизначеними, тобто з них неможливо однозначно знайти компоненти матриць B_k або H_k. Тому існує нескінченно багато способів уточнення (обновлення) цих матриць. Приведемо одну з найбільш поширених , а саме схему Давідона-Флетчера-Пауелла (DFP), яка забезпечує додатну визначеність матриці H_k

$$H_{k+1} = H_k - \frac{H_k y_k y_k^T H_k}{\langle H_k y_k, y_k \rangle} + \frac{s_k s_k^T}{\langle y_k, s_k \rangle} \qquad (\ )$$

тут вираз у дужках $\langle \ \rangle$ позначає скалярний добуток
І тоді вектор квазіньютоновського напрямку d_k обчислюється за простою формулою

$$d_k = -H_k g_k \quad (8.2.5)$$

Пошук оптимальної точки проходить за наступною схемою:

- 1. Вибирається початкове наближення H_0, як правило це одинична матриця I;
- 2. За формулою (8.2.5) обчислюється вектор напрямку пошуку;
- 3. Знаходиться нове наближення $x_{k+1} = x_k + h_k d_k$
- 4. Якщо умова точності ( ) не виконується по ( ) обновлюється
- матриця H_k і обчислення повторюються з пункту 2.

Описаний метод має ще назву - методи змінної метрики .

8.3 Оптимізація при наявності обмежень

Оптимізація при обмеженнях

$$\textit{знайти} \quad \min f(x_1, x_2, ..., x_n,) \tag{8.3.1}$$

$$g_i(x) = 0; \quad i = 1,2,...,m \tag{8.3.2}$$

$$h_j(x) \le 0; \quad \text{j} = 1,2,...,\text{k}, m, k \le n \tag{8.3.3}$$

Умови (8.3.2) і (8.3.3) визначають деяку область у n - вимірному просторі, яка називається допустимою областю, кожна її точка має назву допустимої.

Коли всі співвідношення (8.3.1)-(8.3.3) є лінійними , то це задача лінійного програмування. Якщо цільова функція (8.3.1) квадратична, а функції (8.3.2),(8.3.)- лінійні, то це лінійні задача квадратичного програмування. У випадку коли $f(x), g(x), \text{h}(x)$ є функціями довільного виду, то маємо задачу нелінійного програмування.

На практиці досить часто зустрічаються випадок обмеження .

$$x_k \le \Phi_k \tag{8.3.4}$$

$$x_{m+1} \le x_m - a \tag{8.3.5},$$

які мають назву простих.

Розглянемо задачу з обмеженнями вигляду (8.3.4). Будь-яким ітераційним методом знайдемо рішення задачі без обмежень, в результаті одержимо значення компоненти x_i^*. Тоді наступне наближення для координати x_k знаходиться як

$$x_k = \begin{cases} x_k^*, x_k^* < \Phi_k \\ \Phi_k, x_k^* > \Phi_k \end{cases}$$

У випадку (8.3.5) вводиться нова змінна

$$x_{m+1} = W - a \tag{8.3.6}$$

і також шукається рішення задачі без обмежень, але з новою цільовою функцією

$$f(x_1, x_2, ..., x_{m-1}, W, W-a, x_{m+2}, x_n) \qquad (8.3.7\ )$$

Обмеження (8.3.5) у системі координат $x_m x_{m+1}$ утворює півплощину, а заміною (8.3.6) пошук у допустимій області зводиться до пошуку по її границі, тобто по прямій $x_{m+1} = x_m - a$ (рис.8.5).

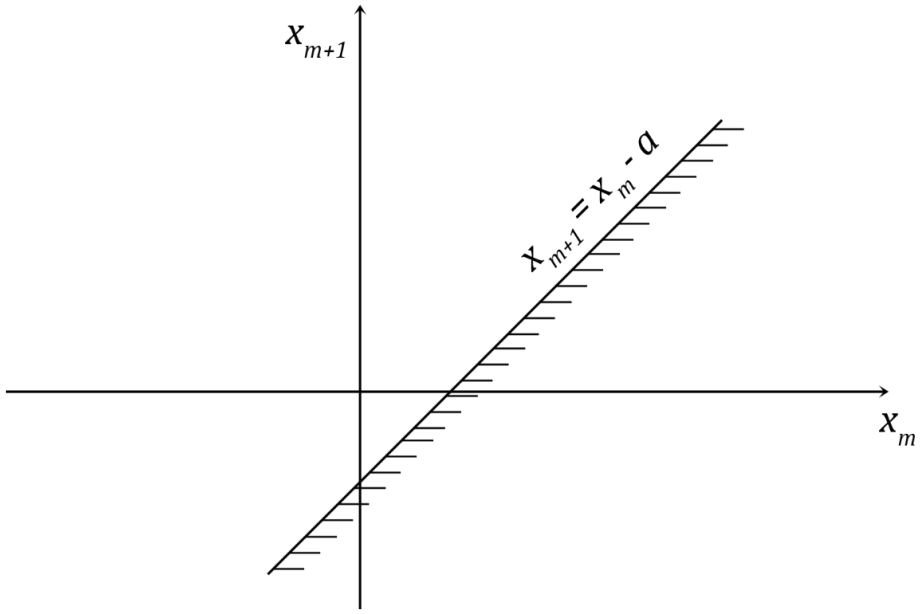


Рис 8.5.- Заміна допустимої на півплощині допустимою прямою.

Метод штрафних функцій є найбільш загальним методом пошуку рішення задачі нелінійного програмування. Ідея метода полягає в зведенні початкової задачі з обмеженнями до послідовності задач безумовної оптимізації. За допомогою функцій, які задають обмеження , будуються так звані штрафні функції, які додаються до цільової функції початкової задачі таким чином, що порушення будь якого обмеження стає невигідним. При цьому обмеження в явному вигляді в задачі оптимізації вже не фігурують. Це забезпечує застосування методів безумовної оптимізації для розв’язку задачі умовної оптимізації . В загальному випадку допоміжна функція має вид

$$\min\left\{f(x)+S_k(x)\middle| x\in R^n\right\}, k=1,2,\ldots$$

Ці задачі, які вже є задачами без обмежень будуються так, щоб

$$x_k \to x^* \in G \text{ при } k_k \to \infty$$

В залежності від положення робочої точки по відношення до допустимою області розрізняють метод зовнішнього штрафу і метод внутрішнього штрафу (метод бар'єрних функцій). У першому випадку штраф не дозволяє точці віддалитися від границі допустимої області, а у другому навпаки покинути її межі.

В залежності від вигляду $S_k(x)$ відрізняють методи внутрішніх і зовнішніх штрафних функцій. Внутрішні штрафні функції так, щоб оптимальні для задачі (8.3.1)- (8.3.3) точки x_k внутрішності G_0 множини G і задовольняли наступним умовам:

1) на більшій частині допустимої множини G штрафні функції близькі до нуля, тобто $f(x)+S_k(x) \to f(x)$ при $k \to \infty$ для всіх з G_0.
2) При наближенні до границі допустимої області G $f(x)+S_k(x) \to \infty$, тому ці функції ще називають бар'єрними (рис.)

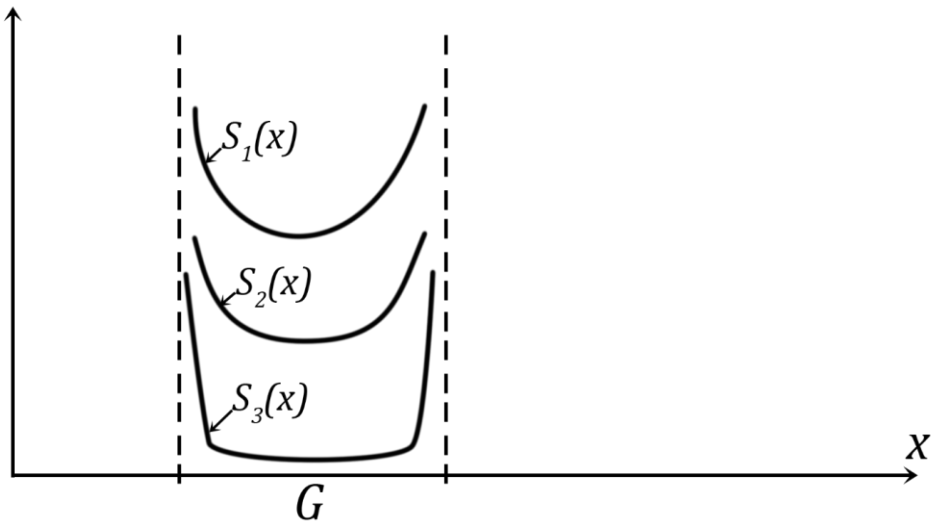


Рис 8.6.- Бар'єрні штрафні функції

ошук мінімуму допоміжних функцій необхідно пожинати з внутрішніх точок допустимої області. При цьому в процесі розв'язку траєкторія спуску ніколи не вийде за межі множини G.

Зовнішні штрафні функції будуються так щоб:

1) У всіх точках допустимої множини G функція $S_k(x) = 0$ всередині і на границі допустимої області або $S_k(x) \to 0$ при $k \to \infty$ для всіх $x \in G$.
2) При виході за межі допустимої області штрафні функції приймають додатні значення і по мірі зростання числа k стають все більш невигідними і зростають тим швидше, чим сильніше порушуються обмеження (рис.).

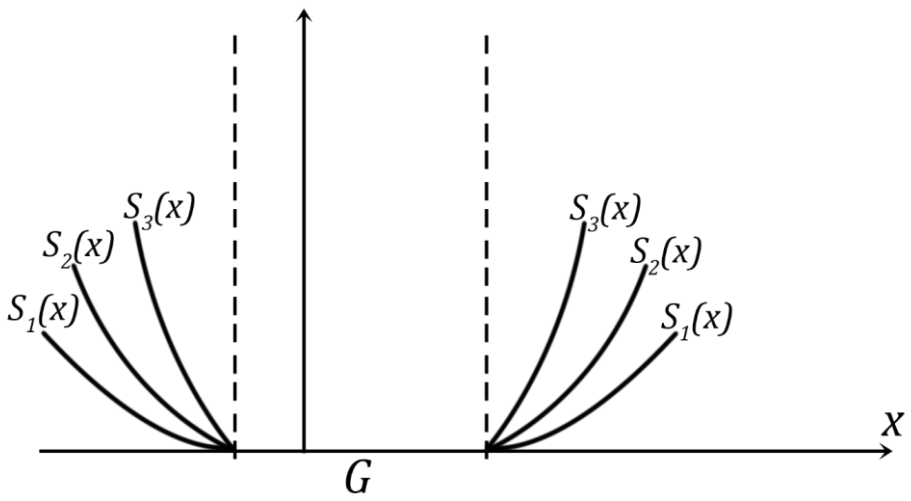


Рис 8.7. Штрафні функції у випадку зовнішнього штрафу.

У випадку зовнішнього штрафу пошук мінімуму допоміжної функції можна починати з довільної точки. У більшості випадків вона буде недопустимою, тому траєкторія спуску може частково бути розташована поза допустимою областю. Якщо мінімум цільової функції розташований на границі допустимої області, то ця траєкторія повністю знаходиться зовні G.

Модифікована цільова функція має вигляд

$$\Phi(\varepsilon_k) = f + S_k$$

Штрафна функція S_k має наступну структуру $S_k = \frac{1}{\varepsilon_k} \cdot F_s$.

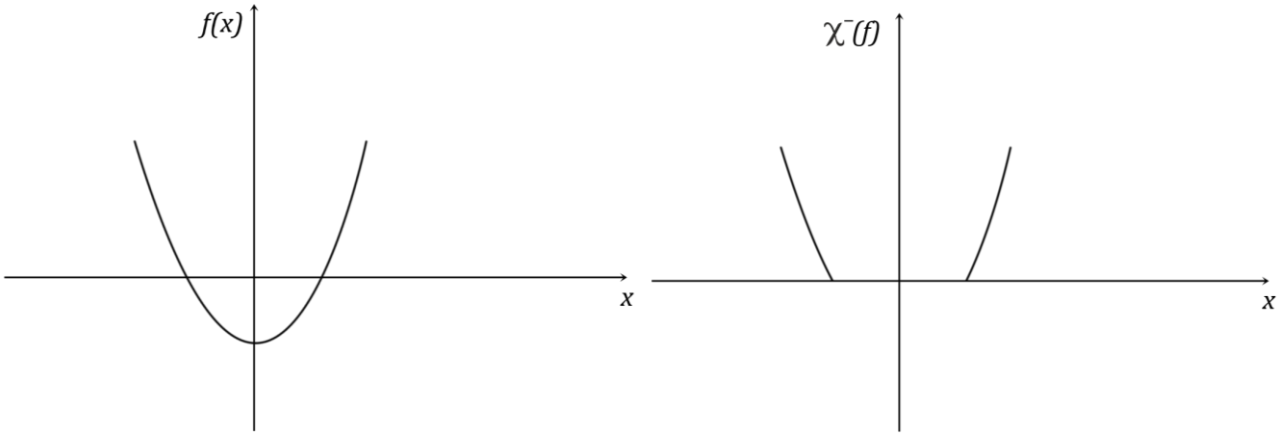
Штрафний тариф ε_k. визначає вклад штрафу у загальній цільовій функції і величина його збільшується по мірі наближення до рішення. Послідовність ε_k будується як $\varepsilon_{k+1} = C_k \varepsilon_k$, $C_k < 1$ і звичайно приймається $C_k = \frac{1}{2} C_{k-1}$.

Функція F_s будується в залежності від конкретного обмеження .

Для обмежень рівностей конструкція штрафної функції очевидна

$$F_s = \sum_{i=1}^{m} (g_i)^2 .$$

У випадку зовнішнього штрафу для обмеження нерівності штрафна функція будується за допомогою функції – зрізки $\chi^-(f) = \max[f, 0]$. Як саме відбувається побудова функції $\chi^-(f)$ видно з рис. 8.8



Початкова функція. Функція зрізка

Рис. 8.8 Формування штрафного додатку для обмежень нерівностей.

і сумарна функція набуває вигляду

$$F_s = \sum_{i=1}^{m} (g_i)^2 + \sum_{j=1}^{k} \chi_j^-$$

Для бар'єрних функцій штрафний додаток може бути зображений в одній з наступних форм

$$F_s = -\sum_{j=1}^{k} \frac{1}{h_j} \quad \text{або} \quad F_s = -\sum_{j=1}^{k} \ln\left|h_j\right|$$

і коли відбувається спроба точки наблизитися до границі допустимої області , тобто $h_i \to 0$, штрафний додаток різко зростає.

ЛАБОРАТОРНА РОБОТА № 10

Рішення задач багатовимірної оптимізації

Завдання до лабораторної роботи. Знайти точку мінімуму функції двох змінних методом Ньютона. Побудувати графік руху до оптимальної точки по ітераціям.

Методичні вказівки до виконання лабораторної роботи.

Для всіх методів багатовимірної оптимізації починаючи з методів першого порядку наступне $(k+1)$ - наближення для оптимальної точки здійснюється за схемою

$$x^{(k+1)} = x^{(k)} + hp^{(k+1)},$$

яка для випадку функції двох змінних у розвернутій формі набуває вигляду

$$x_1^{(k+1)} = x_1^{(k)} + hp_1^{(k+1)},\ x_2^{(k+1)} = x_2^{(k)} + hp_2^{(k+1)},$$

Компоненти вектора напряму пошуку оптимальної точки знаходяться як розв'язок СЛАР другого порядку, коефіцієнти матриці якої і компоненти вектора правої частини визначаються за формулами ( ), тобто

$$a_{11} = \frac{\partial^2 f}{\partial x_1^2}, \quad a_{12} = a_{21} = \frac{\partial^2 f}{\partial x_1 x_2}, \quad a_{22} = \frac{\partial^2 f}{\partial x_2^2},$$

$$b_1 = -\frac{\partial f}{\partial x_1}, \; b_2 = -\frac{\partial f}{\partial x_2}.$$

Створимо для обчислення цільної функції процедуру – функцію f(x,y). Щоб уникнути помилок, що можуть виникнути при визначенні виразів для похідних доцільно скористатися формулами чисельного диференціювання

$$f_i' = \frac{f_{i+1} - f_{i-1}}{2h}, \; f''(x_i) = f_i'' = \frac{f_{i+1} - 2f_i + f_{i-1}}{h^2}.$$

Тоді відповідні коефіцієнти будуть формуватися як виклики

b_1=-0.5*h1*(f(x+h,y)-f(x-h,y));

b_2=-0.5*h1* (f(x,y+h)-f(x,y-h));

a_{11}=h2*(f(x+h,y)-2*f(x,y)+f(x-h,y));

a_{22}=h2*(f(x,y+h)-2*f(x,y)+f(x,y-h));

a_{12}=0.25*h2*(f(x+h,y+h)-f(x+h,y-h)-f(x-h,y+h)+f(x-h,y-h));

Тут h –величина кроку чисельного диференціювання, h=001, а h1:=1/h; h2:=h1*h1.

Для визначення вектора напряму пошуку доцільно організувати за допомогою окремої процедури, заголовок якої має такий вигляд

Procedure Form_Solve(x,y:Real; Var p1,p2:Real);

Тобто вхідними параметрами є координати поточної оптимальної точки, а вихідними - компоненти обчисленого вектора. В тілі процедури виконуються наступні дії:

– обчислення коефіцієнтів матриці і компонентів вектора правої частини СЛАР за за формулами ( );

– обчислення визначників для метода Крамера;

– обчислення компонент p_1і p_2.

Змінні, що відповідають кроку диференціювання, параметрам СЛАР і визначникам у методі Крамера доцільно описати як локальні.

В головній програмі після завдання початкового наближення в циклі Repeat – Until послідовно викликається процедура Form_Solve і виклик процедури одновимірного пошуку, тобто визначення кроку h

Dih(x,y,p1,p2:Real; Var h:Real);

Для формування початкового інтервалу невизначеності, з якого починається робота методу дихотомії ми маємо ліву його точку a=0, і знаємо що для квадратичних функцій величина кроку дорівнює одиниці .Тоді для визначення правої кінцевої точки інтервалу можна застосувати наступний прийом. Починаючи з деякої точки, наприклад a=0,2 рухатись з деяким кроком, наприклад ha =0,2, у напрямку (p_1, p_2) до тих пір поки цільова функція не почне зростати. Програмна реалізація цього вибору має вигляд

```
fc:=f_op(x+p1*a,y+p2*a);
  Repeat c:=a+ha; fa:=fc; a:=c;  fc:=f_op(x+p1*c,y+p2*c);  Until  fc>fa;
  b:=c;
```

При цьому викликається обчислення цільової функції f_op(x,y).

Умова зупинки циклу $\left|x^{(k+1)} - x^{(k)}\right| < \varepsilon$ ( ) у випадку двох вимірів набуває простішого вигляду

$$h\sqrt{p_1^2 + p_2^2} < \varepsilon .$$

Приклад. Знайдемо оптимальну точку цільової функції

$$f(x,y) = (x+1)^2 \cos(\sin(y+1)) + (x-1)^2 y^2 + 20(y-1)^2 (y-3)^2$$

з точністю $\varepsilon = 0.01$. Виберемо початкове наближення $x_0 = 2$, $y_0 = -2$. Значення цільової функції для цього наближення. $f(x_0, y_0) = 4510$. Після виклику процедури Form_Solve маємо. $p_1 = -0{,}5212$ $p_2 = 1{,}2746$, метод дихотомії дає значення кроку $h = 2{,}5$, яке значно більше одиниці, що вказує на те квадратичне наближення у цій точці значно відрізняється від дійсної поведінки функції.

Подальша робота метода відображена нижче

$x_1 = 0.6974$, $y_1 = 1.1860$, $f(x_1, y_1) = 4.3778$, $p_1 = -0.3533$,

$p_2 = -0.2820$, $h_1 = 0.8995$.

$x_2 = 0.3796$, $y_2 = 0.9323$, $f(x_2, y_2) = 1.8557$, $p_1 = -0.1609$,

$p_2 = 0.0576$, $h_2 = 1.0995$

$x_3 = 0.2027$, $y_3 = 0.9957$, $f(x_3, y_3) = 1.5183$, $p_1 = 0.0325$,

$p_2 = -0.0010$, $h_3 = 1.4995$,

$x_4 = 0.2515$, $y_4 = 0.9942$, $f(x_4, y_4) = 1.5157$, $p_1 = -0.0162$,

$p_2 = 0.0005$, $h_4 = 0.6995$.

Після цієї ітерації виконується умова точності і отримуємо наступні координати оптимальної точки $x = 0.2401$, $y = 0.9946$, у якій цільова функція має значення $f(x, y) = 1.5155$.

Контрольні питання.

Скільки разів в межах однієї ітерації обчислюється цільова функція у методі дихотомії і у методі золотого перерізу?

Коли метод Ньютона збігається за одну ітерацію незалежно від початкового наближення?

Для яких цільових функцій задача оптимізації розв'язується прямими методами?

Варіанти завдання до лабораторної роботи.

Варіант	Функція
1	$f = x^4 + 2x^2y^4 - 3y^2 + 12y^4 + 1$
2	$f = x^2 \sin^2(x-2) + x^2y^2 + 12y^4$
3	$f = x^2 \cos^2(x-1) + 6x^2y^2 + y^4$
4	$f = x^6 + 2x^2y^4 + y^4 + 3x^2 - 12$
5	$f = x^6 + 2x^2y^4 + y^4 - 12x^2 + 3$
6	$f = 2x^4 + 6x^4y^2 + y^4 - 12y^2 + 1$

7	$f = x^2 \cos^2(y-1) + 2x^2y^2 + y^4$
8	$f = x^2 \cos^2(y-1) + y^2 \cos^2(x-1)$
9	$f = (x^2-2)\cos^2(y-1) + y^2 \cos^2(x-1)$
10	$f = x^4 + 2y^6 + 3x^2 - 1;$
11	$f = (x-2)^4 + 2y^6 + 4x^2$
12	$f = y^2 \sin^2 y + 2(x-1)^2 y^2 + 12x^4$
13	$f = y^2 \sin^2(y-2) + 2(x-2)^2 y^2 + (x-1)^4$
14	$f = (y-2)(y-1)^2 \sin^2(y-2) + 3x^2y^2 + (x-1)^4$
15	$f = (y-2)^2 + 3x^2y^2 + (x+1)^4$
16	$f = (x-2)^2(y-4)^2 + x^2y^2 + (x+2)^4$
17	$f = (x+3)^2 \cos^2(y-1) + x^2y^2 + (x+2)^4$
18	$f = (y+1)^2 \sin^2(y-1) + (x+2)^2 + x^2(y-3)^2$
19	$f = (y+1)^2 \sin^2(y-1) + (x-5)^2 + x^2(y-3)^2$
20	$f = (x^2-2x+3)\cos^2(y+1) + y^2 \cos^2(x-1)$
21	$f = (1+y+y^2)^2 \cos^2(x-2) + 2x^2(y-1)^2 + 12y^4$
22	$f = (y-1)^2 \sin^4 x + 2(x-1)^2 y^2$
23	$f = (y-1)^2 \cos^2 \sin x + 2(x-1)^2 y^2$
24	$f = (y-2)^2 \cos^2 \sin(x+1) + 2x^2(y+1)^2$
25	$f = y^2 \sin^2 \sin(x+1) + 2x^2(y+1)^2$

ПІСЛЯМОВА

У даному курсі розглянуто основні методи чисельного аналізу. При цьому крім формального запису метода робиться акцент на його ідею і границі застосування. Значну увагу приділено наближенню функцій по скільки ця задача є своєрідним фундаментом як для прикладних задач, так і для розробки нових методів. Досить детально, також розглянуто скінчено різницевий підходу до рішення диференціальних рівнянь, який дозволяє звести вихідну задачу до більш простих, наприклад СЛАР, а в окремих випадках до простих рекурентних формул.

Для рішення будь якої задачі існує декілька методів що створює проблему вибору. Щоб обрати який метод використати для рішення даної конкретної задачі бажано знати яким чином він одержаний і які спрощення при цьому приймались. В даний час існує багато пакетів програм (Math Cad, Maple), які є зручними для рішення однієї конкретної задачі. Але коли задача складається з декілька етапів більш ефективним є створення власної програми . При цьому дуже корисно врахувати специфіку задачі і досить часто

використання більш простого метода є ефективнішим. Так в задачі багатовимірної оптимізації у випадку простих обмеженнях шляхом невеликої модифікації метода по координатного спуску результат можна досягнути швидше ніж за допомогою універсального метода штрафних функції

Методи з'являються постійно, але потрібен певний час для їх апробації, щоб довести на практиці переваги перед існуючими. Якщо існує альтернатива вибору з двох методів один з яких новий, а другий використовується досить довго, то перевага буде на боці останнього.

І якщо ще кілька років тому, найважливішими критеріями вибору були об'єм обчислень і пам'яті, то при сучасних, практично без граничних ресурсах обчислювальної техніки, на перші місця виходять точність і стійкість.

Проблема збіжності і стійкості у багатьох випадках може бути розв'язана тільки шляхом чисельного експерименту тобто аналізу поведінки рішення в залежності від числа базисних функцій в його розкладенні, кількості вузлів інтегрування, параметрів різницевої сітки або скінчених елементів.

Такі технічні прийоми як нормування області пошуку рішення і перехід до безрозмірних параметрів і змінних є обов'язковими на етапі переходу від математичної моделі до чисельного розв'язку. Вони дозволяють повисити точність обчислювань як у випадку чисельного диференціювання та інтегрування, а також порівняти результати одержані різними методами і різними авторами.

ЛІТЕРАТУРА

Н.С. Бахвалов, Н.П. Жидков, Г.Н. Кобельков Численные методы. М.: Наука, 1987-600с.

Ф. Гилл, У. Мюррей, М. Райт Практическая оптимизация. Пер. с англ.-М.:Мир,1985-509с.

В.И. Крылов, В.В. Бобков, П.И. Монастырный Вычислительные методы вышей математики.Т1. Мн., Вышэш. школа.1972-584с.

А.А. Самарский, А.В. Гулин Численные методы. М.: Наука, 1989-432с.

Дж. Форсайт, М. Малькольм, К. Моулер Машинные методы математических вычислений. Пер. с англ.-М.:Мир,1980-279с.

Т. Шуп Решение инженерных задач на ЭВМ: Практическое руководство. Пер. с англ.-М.:Мир,1982-238с.