К. М. Рудаков

Чисельні методи аналізу
в динаміці та міцності конструкцій

К. М. Рудаков

Чисельні методи аналізу

в динаміці та міцності конструкцій

Видання друге, виправлене та доповнене

Навчальний посібник для студентів вищих навчальних закладів, які навчаються за освітньою програмою "Динаміка і міцність машин" спеціальності 131 Прикладна механіка

КИЇВ – 2025

УДК 519.6 : 517.9 : 539.3

Гриф на перше видання надано Міністерством освіти і науки України (лист № 1.4/18-Г-1425 від 28.12.2006 р.)

Рецензенти: д.т.н., проф. *В.Г. Піскунов* (Національний транспортний університет МОН України); д.т.н., проф. *М.К. Кучер* (Інститут проблем міцності ім. Г.С. Писаренка НАН України); д.т.н., с.н.с. *А.З. Галішин* (Інститут механіки ім. С.П. Тимошенка НАН України); д.т.н., проф. *В.Ф. Оробей* (Одеський Національний політехнічний університет МОН України)

Рудаков К.М.

Чисельні методи аналізу в динаміці та міцності конструкцій : Навч. посібник [для студ. вищ. навч. закл. Електронний ресурс] / К.М. Рудаков – Київ, 2025. – 490 с.

Розглядаються методи та алгоритми чисельного розв'язування крайових задач механіки твердого деформівного тіла.

У першій та другій частинах розглядаються супутні проблеми: наближені числа та обчислення, основи теорії операторних рівнянь і функціонального аналізу, а також чисельні методи алгебри: знаходження коренів рівнянь, розв'язування систем рівнянь, чисельне інтерполювання, наближення, диференціювання та інтегрування функцій, інтегрування звичайних диференційних рівнянь та їх систем, тощо.

У наступних частинах розглядаються постановки, методи та алгоритми розв'язування класичних крайових задач: теплопровідності, термопружності, термопружнопластичності та повзучості, динамічної термопружності, про знаходження власних форм та частот коливань пружних тіл, про контактні взаємодії.

Для студентів вищих технічних навчальних закладів, які спеціалізуються з питань міцності конструкцій. Може бути корисним викладачам, аспірантам, науковим працівникам та інженерам, які застосовують методи математичного моделювання для чисельних розрахунків на ЕОМ теплового, напружено-деформованого і динамічного стану елементів конструкцій та машин.

УДК 519.6 : 517.9 : 539.3

Зміст

Частина I
ВСТУП ДО ЧИСЕЛЬНИХ МЕТОДІВ

Частина II
ЧИСЕЛЬНІ МЕТОДИ АЛГЕБРИ

Частина IV
АЛГОРИТМИ МЕТОДУ СКІНЧЕННИХ РІЗНИЦЬ

Частина V
МЕТОД ЗВАЖЕНИХ ПОХИБОК НАБЛИЖЕННЯ ТА ІНШІ МЕТОДИ АЛГЕБРАЇЗАЦІЇ КРАЙОВИХ ЗАДАЧ

Частина VII

АЛГОРИТМИ МЕТОДУ СКІНЧЕННИХ ЕЛЕМЕНТІВ ПРИ РОЗВ'ЯЗУВАННІ СТАЦІОНАРНИХ І ЕВОЛЮЦІЙНИХ ЗАДАЧ

Частина VIII

АЛГОРИТМИ МЕТОДУ СКІНЧЕННИХ ЕЛЕМЕНТІВ ПРИ РОЗВ'ЯЗУВАННІ КРАЙОВИХ ЗАДАЧ ДИНАМІКИ

До другого видання

Перше видання цієї книги вийшло у 2007 році. Вона була написана на основі лекцій, які автор читав студентам спеціальності "Динаміка і міцність машин" у Національному технічному університеті України "Київський політехнічний інститут" (НТУУ "КПІ").

З тих часів у власно чисельних методах аналізу в динаміці та міцності конструкцій мало що змінилося. Але змінилися деякі погляди автора на актуальність тих або інших методів та алгоритмів. І багато що змінилося у країні та навчальному процесі. НТУУ "КПІ" змінив назву на Національний технічний університет України "Київський політехнічний інститут імені Ігоря Сікорського". В Україні змінився перелік спеціальностей, і тепер ця тематика викладається для освітньої програми "Динаміка і міцність машин", яка існує в рамках спеціальності 131 Прикладна механіка. Значно зменшився час, що виділяється на лекції, але збільшився на самостійну роботу. Змінилися вимоги до оформлення посібників. Були виявлені деякі прикрі помилки у тексті та формулах. Окрім того, з'явилася така форма видання посібників (і не тільки), як електронна.

У цьому виданні враховані вагомі зміни, виправлені помилки, додано матеріал у декількох розділах, та навіть цілі розділи.

* * *

Повний об'єм здобутків у чисельних методах є практично неосяжним, тому фрагментарність викладеного в книзі матеріалу неминуча. Не розглядалося багато методів та розділів, які не є домінуючими або потребують дуже багато місця. Ще враховувалося основне призначення цієї книги: для студентів (аспірантів) технічних університетів.

Частина І
ВСТУП ДО ЧИСЕЛЬНИХ МЕТОДІВ

Передмова

Процес розвитку техніки висуває перед фахівцями з міцності завдання щодо підвищення надійності та довговічності машин і конструкцій, працюючих у складних умовах експлуатації. Це потребує велику кількості розрахунків. При цьому розглядається суцільне середовище або комбінація середовищ (тверде, рідке, газоподібне) у взаємодії; їхній стан моделюється рівняннями, лінійними або нелінійними, а розв'язки дозволяють отримати числові значення розподілу полів температур, переміщень, деформацій, напружень, власних частот коливань та ін., що, в свою чергу, є інформацією для оцінки міцності, жорсткості, довговічності машин, елементів конструкцій і агрегатів у цілому.

Аналітичні розв'язки можливі не завжди, зазвичай – для тіл канонічних обрисів. Застосовування наближених методів майже неминуче.

Історично однієї з перших задач наближених (чисельних) методів була задача про квадратуру круга, інакше – задача про значення числа π. Площа круга $A = \pi R^2$, при радіусі $R = 1$ значення $\pi = A$, а A обчислювали через суму площ геометричних фігур, вписаних в коло, збільшуючи їхню кількість. Ще древні арабські математики визначили число π з точністю до 40 знаків.

Чисельні методи – це інтерпретації математичних моделей для їхньої реалізації за допомогою простих математичних дій: додавання, віднімання, множення, ділення, а також логічних операцій: "так", "ні", "і", "або" й операції порівняння "більше", "менше", "дорівнює". З числами саме це, і тільки це, але дуже швидко вміють робити електронні обчислювальні машини (ЕОМ). І саме тому з появою ЕОМ чисельні методи отримали велике прискорення у своєму розвитку.

Ще одна важлива обставина: ЕОМ є пристроєм з дискретним числовим рядом. Причина – обмежена кількість розрядів, що виділяються для запису будь-якого числа.

З цих двох причин в ЕОМ будь-яка розрахункова модель повинна бути представлена *дискретно* та *алгебраїзована*: безперервність простору й часу замінена дискретним описом, тобто набором геометричних точок (*вузлів*) і *часових шарів*, а всі не алгебраїчні математичні об'єкти (наприклад, інтеграли й диференціали) – замінені з достатньою точністю на алгебраїчні наближення.

Математична модель як *крайова задача* – це сукупність заданих і розшукуваних фізичних величин, між якими сформульовані математичні (алгебраїчні, диференціальні, інтегральні), та логічні зв'язки, зокрема й обме-

жуючі в просторі та часі. Ці зв'язки можуть бути як лінійними, так і нелінійними. Крайова задача повинна мати хоча би один розв'язок. Постановкою крайових задач, пов'язаних з міцністю, й створенням методів розв'язування їх займаються: математична фізика, теорія пружності, теорія пластичності та повзучості, гідроаеродинаміка, теорія коливань, механіка тріщин, інші наукові дисципліни. Інколи на практиці потрібно розв'язувати задачі, для яких немає повної чіткості в постановці: наприклад, не доведено теореми про існування чи однозначності розв'язку (однозначність розв'язку – не обов'язкова вимога, оскільки є крайові задачі, які мають багато розв'язків, наприклад, задачі втрати пружної стійкості, задачі про власні частоти та форми коливань).

При розв'язуванні крайових задач на ЕОМ дискретизація простору й часу, у сполученні з алгебраїзациею, призводить до появи *систем алгебраїчних рівнянь* (САР, лінійних або нелінійних, однієї або багатьох) щодо значень характеристик у вузлах (*вузлових функцій*) дискретного простору та на часових шарах. Після розв'язування САР на основі отриманих вузлових значень параметрів, при необхідності, обчислюються інші параметри.

Доказано, що для підвищення точності дискретної моделі (порівняно з неперервною моделлю) потрібно зменшувати відстані між вузлами (часовими шарами) з одночасним збільшенням кількість вузлів та часових шарів.

З появою ЕОМ та подальшим розвитком чисельних методів з'явився потужний і зручний метод теоретичного дослідження складних об'єктів та процесів, які допускають математичне моделювання, – *обчислювальний експеримент*.

Етапи проведення обчислювального експерименту:

1 – створення нової математичної моделі або вибір математичної моделі, що вже існує, *постановка задачі*. Дослідження коректності математичної моделі, встановлення меж її застосування, існування та єдиного варіанта розв'язку, знаходження окремих і, якщо це можливо, аналітичних розв'язків, підбір тестів для майбутньої перевірки якості обчислювальних алгоритмів;

2 – побудова (або вибирання) чисельного методу розв'язування сформульованої математичної моделі, складання обчислювального алгоритму;

3 – програмування алгоритму для ЕОМ, його тестування;

4 – створення розрахункової моделі актуального об'єкта;

5 – проведення розрахунків із застосуванням ЕОМ;

6 – аналіз, документування отриманих результатів і, при необхідності, уточнення математичної моделі.

Проведення обчислювального експерименту щодо складних технічних систем (наприклад, ракет, літаків, турбін і їх елементів) у повному обсязі дозволяє значно знизити витрати часу та інших ресурсів при їхньої розробці та експлуатації. Тобто обчислювальний експеримент – це інструмент пізнання.

Проблемами другого етапу обчислювального експерименту займається теорія чисельних методів – один з найбільших розділів прикладної математики.

Під словом *алгоритм* розуміють набір інструкцій, що описують порядок дій виконавця для розв'язування задачі за кінцеву кількість дій.

Під *обчислювальним алгоритмом* (ОА) розуміють встановлену послідовність математичних операцій і операцій логіки, яка приводить до отримання розв'язку задачі. Очевидна багатоваріантність ОА для кожної конкретної задачі. Вимагається, щоб ОА був достатньо точним і швидким. Ці загальні вимоги до ОА – суперечливі, оскільки точність розв'язку крайової задачі:

- залежить від щільності дискретизації області, в якій задача розглядається: дрібніша дискретизація підвищує точність, але і ресурсів ЕОМ вимагається більше;
- пов'язана з обмеженою точністю внутрішнього представлення чисел в ЕОМ, наявністю "машинної" нескінченності. Теоретично стійкий обчислювальний алгоритм на ЕОМ може виявитися нестійким внаслідок надмірного накопичення похибок округлення;
- залежить від кількості операцій обчислювального алгоритму, потужності ЕОМ і якості програми, тобто від часу, виділеного на розв'язування задачі.

При розробці обчислювального алгоритму кінцева мета – виділення з деякої множини алгоритмів оптимального ОА шляхом поступового залучення вимог якості: наближення, коректності, стійкості, економічності тощо.

Задача поставлена коректно для ЕОМ, і алгоритм теж є коректним, якщо:

- задача має обмежену кількість розв'язків при будь-яких значеннях вхідних даних з обумовленого діапазону;
- розв'язки безперервно залежать від вхідних даних (алгоритм є стійким).

Загальними принципами побудови наближень є:

- принцип однорідності (одноманітності);
- принцип консервативності (відображення на сітці деякого закону збереження, балансу).

Створення обчислювальних алгоритмів для розв'язування *крайових* задач – не єдина проблема чисельних методів. Є багато інших проблем, зокрема таких, що виконують допоміжні функції при проведенні обчислювального експерименту.

Саме тому в книзі розглядаються такі напрямки першого та другого етапу проведення обчислювального експерименту:

- створення різноманітних апроксимацій, інтерполяцій та екстраполяцій, диференціювання та інтегрування, тощо;
- побудова дискретних наближень, дослідження характеристик їх якості;
- побудова методів розв'язування отриманих наближень, дослідження характеристик якості цих методів;
- побудова алгоритмів розв'язування отриманих рівнянь й їхніх систем прямими або ітераційними методами.

Розділ 1

НАБЛИЖЕНІ ЧИСЛА ТА ОБЧИСЛЕННЯ

1.1. Наближені числа. Похибки, їхні джерела

1.1.1. Різновиди похибок наближеного числа, їхні джерела

Наближене число a – число, що незначно (на величину похибки) відрізняється від точного A та заміняє його при проведенні розрахунків.

В таблиці 1.1. наведені різновиди похибок наближеного числа $a \approx A$, визначеність, відповідні позначення та математичні вирази.

Таблиця 1.1. Різновиди похибок наближеного числа

Різновид похибок числа a	Визначеність	Позначення, вираз
похибка	реальна	$\Delta a = a - A$
абсолютна похибка		$\Delta = \lvert \Delta a \rvert = \lvert a - A \rvert$
нижня границя похибки		$\underline{\Delta} a$
верхня границя похибки		$\overline{\Delta} a$
максимальна похибка		$\overline{\Delta} a = \max\{\lvert \underline{\Delta} a \rvert, \lvert \overline{\Delta} a \rvert\}$
відносна похибка		$\delta = \Delta / \lvert A \rvert$ або $\delta = \Delta / \lvert a \rvert$
гранична абсолютна похибка	оціночна	$\Delta_a \geq \Delta$
гранична відносна похибка		$\delta_a \geq \delta$

Точне значення A та наближене значення a часто пов'язують формулою

$$a = A \cdot (1 + \varepsilon), \qquad (1.1)$$

де ε – відносна похибка обчислень δ, але з урахуванням знаку.

Розрізняють декілька джерел похибок обчислень. Їхні назви та причини виникнення зведені до таблиці 1.2.

Таблиця 1.2. Основні джерела похибок обчислень, причини виникнення похибок

№	Джерело похибок	Причини виникнення похибок, пояснення
1	математична модель (постановка задачі)	точна постановка задачі невідома, є спрощення
		точна постановка задачі відома, але розв'язати її точно неможливо або недоречно
2	вихідні (початкові) дані	точні значення невідомі
3	метод (алгоритм)	метод (алгоритм) є наближеним
4	дискретизація (простору, часу)	безперервний простір та/або час замінюється на дискретний набір точок (вузлів), моментів часу
5	нестійкий алгоритм	малі вихідні похибки при діях підсилюються і накопичуються лавиноподібним чином
6	усікання	переривання нескінченних процесів (ітерацій, рядів)

Продовження таблиці 1.2.

7	дії з наближеними числами	похибки при діях можуть накопичуватися
8	округлення	використовується скінченна кількість знаків для представлення кожного "довгого" числа
9	хиби	грубі людські помилки

Приклад похибок вихідних даних: точні значення "фізичних констант", модуля Юнга, числа e (основа логарифма), числа π не відомі.

Для виявлення грубих людських помилок (хиб) при введенні даних в ЕОМ рекомендують застосовувати процедури їх контролю.

Є й інша класифікація похибок та їхніх зв'язків з джерелами (за виключенням п.9 – хиб). Позначимо: A – точне (реальне) значення величини, що описує деякий процес; $\tilde{a}$ – точний розв'язок математичної моделі процесу; $\underset{\sim}{a}$ – наближений розв'язок моделі процесу, але у припущенні, що при обчисленнях не проводилися округлення; a^* – реальний результат обчислень з округленнями. Тоді можна сформувати таблицю 1.3.

Таблиця 1.3. Похибки обчислень, їхні зв'язки з джерелами похибок

№	Назва похибки	Формула	П.п. таблиці 1.2
1	"неусувна" похибка	$\tilde{\Delta} = \tilde{a} - A$	1, 2
2	похибка методу	$\underset{\sim}{\Delta} = \underset{\sim}{a} - \tilde{a}$	3 … 7
3	похибка обчислень	$\Delta^* = a^* - \underset{\sim}{a}$	8
4	повна похибка	$\Delta = \tilde{\Delta} + \underset{\sim}{\Delta} + \Delta^* = a^* - A$	1 … 8

Всі ці похибки вдається хоча б частково зменшити, запропонувавши більш вдалу математичну модель (п.1), метод (п.2) та/або точність збереження даних (п.3 таблиці 1.3). Але часто можна навіть збільшувати можливі похибки майбутнього розв'язку, якщо насправді велика точність не потрібна.

Практична рекомендація: похибки методу повинні бути на декілька порядків меншими, ніж "неусувні" похибки, а похибки обчислень – значно меншими, ніж усі інші похибки.

Похибка, яка прогнозується ще до розв'язування задачі, називається *апріорною*. На основі отриманого розв'язку задачі одержують *апостеріорну* оцінку похибки.

1.1.2. Форми запису чисел. Значуща цифра. Вірні знаки

Будь-яке дійсне число a можна представити в позиційній формі запису як:

$$a = \pm(\alpha_m \cdot b^m + \alpha_{m-1} \cdot b^{m-1} + \alpha_{m-2} \cdot b^{m-2} + ... + \alpha_{m-n} \cdot b^{m-n}), \qquad (1.2)$$

де $\pm$ – знак числа; $b > 1$ – основа системи числення; m – старший розряд числа a; α_i – розряди, які є цифрами з базисного набору системи (наприклад: 0, 1, 2, 3, 4, 5, 6, 7, 8, 9 – десяткової системи числення; 0, 1 – двійкової); $n+1$ – кількість значущих розрядів (цифр). Якщо $m - n \geq 0$, то це – ціле число, $m - n < 0$ – не ціле. Такий варіант представлення чисел має форму числа з *фіксованим* розділовим знаком.

Приклад. У десятковій системі числення число $a = 84.902 = 8 \cdot 10^1 + 4 \cdot 10^0 + 9 \cdot 10^{-1} + 0 \cdot 10^{-2} + 2 \cdot 10^{-3}$, тобто для нього $m = 1$, а $n + 1 = 5$.

Інша форма представлення дійсних чисел: із *плаваючим* розділовим знаком:

$$a = \pm M \cdot b^q, \tag{1.3}$$

де M – мантиса числа; $\pm$ – знак числа; q – ціле число. Якщо $b = 10$, то таку форму ще називають математичною, а замість $\cdot 10$ з метою скорочення запису часто пишуть e. Якщо $b^{-1} \le M < 1$, то така мантиса зветься *нормалізованою*, а весь запис – нормалізованим.

Приклад. В десятковій системі числення число $a = -0.739 \cdot 10^{-5} = -0.739e-5$, тобто для нього знак числа "мінус", мантиса $M = 0.739$, а $q = -5$.

Значущий розряд (*значуща цифра*) наближеного числа – це одна з цифр S на будь-якій позиції ($0 < S < b$), та цифра 0, якщо остання розташована між значущими цифрами S або є представником збереженого розряду. Всі інші цифри 0 (нулі) вказують лише на розряди числа.

Ще варіант визначення: *значущі розряди* (*значущі цифри*) наближеного числа – це всі цифри в числі, починаючи з першої ненульової цифри зліва.

Приклад. У наближеному числі 0.00<u>3001900</u> (десяткова система числення) значущі цифри підкреслені, причому число 0.0030019 йому не рівноцінне. Останні два нулі у числі 0.003001900 вказують на те, що ці цифри значущі (там точно – нулі).

У числі <u>5039000</u> останні три нулі вказують на те, що в ньому є такі розряди (сім значущих цифр). Але такі числа можна представляти як, наприклад, $5.039 \cdot 10^6$ (чотири значущих цифри), або $5.039000 \cdot 10^6$ (сім значущих цифр). Тобто у ньому є як мінімум чотири значущих цифри (5039). Збереження нулів наприкінці числа ($5.039000 \cdot 10^6$) теж вкаже на те, що ці цифри значущі.

Вірні знаки – це k перших значущих цифр числа, якщо абсолютна похибка цього числа не перевищує *половини* (інший варіант – *цілої*) одиниці k-го розряду. Як знайти k? Для цього застосовують математичний запис визначення кількості вірних знаків:

$$\Delta = |a - A| \le 0.5 \cdot b^{m+1-k}, \tag{1.4}$$

де m – старший розряд числа a.

Приклад. $\Delta = |99.96 - 100| = 0.04 < 0.05 = 0.5 \cdot 10^{-1} = 0.5 \cdot 10^{m+1-k}$. Оскільки тут $m = 1$; $1 + 1 - k = -1$, то $k = 3$, тобто число 99.96 має три вірних десяткових знака.

У всіх таблицях, що приводяться в довідниках, всі значущі цифри повинні бути вірними, тобто відповідати умові (1.4).

Коли записують абсолютну або відносну похибку, зазвичай використовують *одну* або *дві* значущих цифри, не більше.

Якщо число A записують у формі

$$A = a \pm \Delta_a, \tag{1.5}$$

то числа a та Δ_a прийнято записувати з *однаковою* кількістю знаків *після* розділового знаку (якщо він є).

1.1.3. Округлення чисел

У десятковій системі числення *правила округлення чисел* до n значущих цифр (за принципом доповнення) є такими: відкидаються всі цифри, що стоять правіше, причому, якщо перша з відкинутих цифр:

а/ < 5, то результат одержано;

б/ > 5, то остання цифра з цифр, що залишилися, збільшується на одиницю;

в/ = 5, а серед відкинутих цифр були не нулі, то остання цифра з цифр, що залишилися, збільшується на одиницю;

г/ = 5, а серед відкинутих цифр всі були нулі, то остання цифра з цифр, що залишилися, повинна бути парною (непарна збільшується на одиницю).

Точність наближеного числа залежить від кількості *вірних* значущих цифр.

Практичне правило наближених обчислень: кількість значущих цифр в проміжних результатах не повинна перевищувати кількості вірних цифр більш ніж на 1 або 2 одиниці, а в остаточних – на 1 одиницю. Якщо при цьому абсолютна похибка результату для останнього збереженого розряду не перевищує 2, то останню цифру називають *сумнівною*. Тобто сумнівними є такі останні цифри в результатах наближених обчислень: 1, 2, 8, 9.

1.1.4. Зв'язок граничної відносної похибки наближеного числа з кількістю вірних знаків

Такий зв'язок встановлено доведеною теоремою: якщо наближене число $a > 0$ має k вірних знаків, то його гранична відносна похибка

$$\delta_a \le \frac{1}{\alpha_m} \cdot b^{1-k} \text{ при } k = 1; \quad \delta_a \le \frac{1}{2\alpha_m} \cdot b^{1-k} \text{ при } k > 1, \tag{1.6}$$

де α_m – перша значуща цифра числа a. Друга формула (1.6) є наближеною, але її точність швидко підвищується з ростом b та k, оскільки в знаменнику було прийнято $2\alpha_m - b^{-k} \approx 2\alpha_m$ (знехтували величиною b^{-k}).

Приклад. Яка гранична відносна похибка числа a = 3.14, яке часто застосовують замість числа π? Тут $b = 10$, $\alpha_m = 3$ та k = 3. Тому відповідно до другої формули (1.6) $\delta_a \le 10^{1-3} / (2 \cdot 3) = 1/600 \approx 0.001(6)$. Тобто відносна похибка числа a = 3.14 в порівнянні з числом π складає менш ніж 0.2 відсотка.

Приклад. Знайти, скільки вірних знаків достатньо зберегти при обчисленні $a = \sqrt{20}$ з точністю не гірше 0.1 відсотка. Отже, $\delta_a \le 0.001$. Оскільки $\sqrt{20} \approx 4.472136...$, то $\alpha_m = 4$. Згідно з другою формулою (1.6): $10^{1-k}/(2\alpha_m) = 10^{1-k}/(2 \cdot 4) = 0.125 \cdot 10^{1-k} \le 0.001$. Вочевидь, для цього потрібно мати $k = 4$. Отже, $a = \sqrt{20} \approx 4.472$ має 4 вірних знака та відносну похибку $\delta_a \le 0.001$.

1.2. Оцінки похибок при наближених обчисленнях

1.2.1. Оцінки похибок при елементарних обчисленнях

Похибки суми наближених чисел. Точне значення суми декількох чисел можемо записати як

$$A = \pm A_1 \pm A_2 \pm ... = (\pm a_1 \pm \Delta a_1) + (\pm a_2 \pm \Delta a_2) + ... = \sum_i \pm a_i + \sum_i \pm \Delta a_i = a \pm \Delta a .$$

Тоді $\Delta_a \le \sum_i |\Delta a_i| = \sum \Delta_{a_i}$. Тобто абсолютну похибку суми визначають найменш точні члени суми, а за рахунок найбільш точних членів суми суттєво підвищити точність не вдається.

Якщо всі $A_i > 0$, то відносна похибка $\delta = \left(\sum \Delta_{a_i}\right) / A = \left(\sum \left(\delta_{a_i} A_i\right)\right) / A$. Якщо позначимо $\overline{\delta} = \max\{\delta_{a_1}, \delta_{a_2}, ...\}$, то $\delta_a \le \left(\overline{\delta} \sum A_i\right) / A = \overline{\delta}$.

Отже, граничні похибки суми наближених чисел визначаються виразами (замість індексу a запишемо індекс суми Σ):

$$\Delta_\Sigma \le \sum \Delta_{a_i} ; \qquad (1.7\text{-а})$$

$$\delta_\Sigma \le \Delta_\Sigma / |a| \quad \text{та} \quad \delta_\Sigma \le \max\{\delta_{a_1}, \delta_{a_2}, ...\}, \text{ якщо всі } A_i > 0 \text{ або всі } A_i < 0 . \qquad (1.7\text{-б})$$

Похибки різниці двох наближених чисел визначаються виразами, які є окремим випадком з (1.7):

$$\Delta_{a_1 - a_2} \le \sum \Delta_{a_i} ; \quad \delta_{a_1 - a_2} = \Delta_{a_1 - a_2} / |a_1 - a_2| , \quad \text{якщо } a_1 \ne a_2 . \qquad (1.8)$$

Тому, якщо $a_1 \approx a_2$, то відносна похибка $\delta_{a_1 - a_2}$ може бути дуже великою. Про це говорять як про "втрату точності при розрахунку різниці", а саме таке явище називають "катастрофічним взаємним винищенням".

Похибка добутку наближених чисел. Якщо серед наближених чисел немає такого, що дорівнює нулю, а результат позначено як a, то добуток n чисел: $A = A_1 \cdot A_2 \cdot ... \cdot A_n = (a_1 + \Delta a_1) \cdot (a_2 + \Delta a_2) \cdot ... \cdot (a_n + \Delta a_n) \approx (a_1 \cdot a_2 \cdot ... \cdot a_n) + (a_2 \cdot a_3 \cdot ... \cdot a_n)\Delta a_1 + (a_1 \cdot a_3 \cdot ... \cdot a_n)\Delta a_2 + ... + (a_1 \cdot a_2 \cdot ... \cdot a_{n-1})\Delta a_n = a + \Delta a$, де $a = a_1 \cdot a_2 \cdot ... \cdot a_n$. Тут відкинути як члени інших порядків малості члени з добутками похибок. Відносна похибка $\delta = \Delta a / a \approx \Delta a_1 / a_1 + \Delta a_2 / a_2 + ... + \Delta a_n / a_n = \pm\delta_{a_1} \pm \delta_{a_2} \pm ... \pm \delta_{a_n} = \sum_{i=1}^{n} (\pm \delta_{a_i})$. Тому граничні відносна та абсолютна похибки добутку наближених чисел

$$\delta_a \le \sum \delta_{a_i} ; \quad \Delta_a = |a| \cdot \delta_a . \qquad (1.9)$$

Приклад. Є два наближених числа a_1=12.2; a_2=73.56, в яких всі знаки вірні. Знайти добуток цих чисел, тобто числа $a = a_1 \cdot a_2$, та його похибки.

Оскільки всі знаки – вірні, то абсолютні похибки чисел не перевищують половини останнього розряду, тобто: Δ_{a_1} = 0.05; Δ_{a_2} = 0.005; δ_{a_1} = 0.05/12.2; δ_{a_2} =0.005/73.56; $\delta_a \le \sum \delta_{a_i}$ =0.05/12.2+0.005/73.56 ≈ 0.0042. Обчислимо із збереженням всіх знаків $a = a_1 \cdot a_2 = 12.2 \cdot 73.56 = 897.432$. Таким чином, $\Delta_a = |a| \cdot \delta_a = 897.432 \cdot 0.0042 \approx 3.6$. Висновок: число a має лише два вірних знака, тому $a = a_1 \cdot a_2 = 12.2 \cdot 73.56 \approx 897 \pm 4$.

Окремий випадок. Якщо обчислюється $a = n \cdot c$, де число n – точне, а число c – наближене, то $\delta_a = \delta_b$, а $\Delta_a = |a| \cdot \delta_a = |n \cdot c| \cdot \delta_b = |n| \cdot |c| \cdot \delta_b = |n| \cdot \Delta_b$. Висновок: абсолютна похибка змінилася у n разів.

Похибки частки двох наближених чисел. Якщо $a = a_1 / a_2$ й $a_2 \neq 0$, то аналогічно похибкам добутку наближених чисел можна отримати, що:

$$\delta_a \leq \sum \delta_{a_i}; \quad \Delta_a = |a| \cdot \delta_a. \tag{1.10}$$

Приклад. Є два наближених числа a_1=25.7; a_2=3.6, в яких всі знаки є вірними. Знайти число $a = a_1 / a_2$ та його похибки.

Оскільки всі знаки – вірні, то абсолютні похибки чисел не перевищують половини останнього розряду, тобто: Δ_{a_1}=0.05; Δ_{a_2}=0.05; δ_{a_1}=0.05/25.7; δ_{a_2}=0.05/3.6; $\delta_a \leq \sum \delta_{a_i}$=0.05/25.7+0.05/3.6$\approx$0,0158. Обчислимо із збереженням всіх знаків $a = a_1 / a_2 = 25.7/3.6 \approx 7.13(8)$. Таким чином, гранична абсолютна похибка $\Delta_a = |a| \cdot \delta_a = 7.13(8) \cdot 0.0158 \approx 0.11$. Висновок: число a має лише один вірний знак, тому $a = a_1 / a_2 = 25.7/3.6 \approx 7.1 \pm 0.1$.

Відносна похибка степені. Якщо $a = c^m$, причому $c \neq 1$, а m є точним дійсним числом, то з виразів для похибки добутку наближених чисел:

$$\delta_a = m \cdot \delta_c; \quad \Delta_a = |a| \cdot \delta_a. \tag{1.11}$$

Приклад. Є наближене число c=32.9, в якому всі знаки є вірними. Знайти число $a = \sqrt[n]{c}$ та його похибки, якщо число n=2.5 є точним.

Запишемо $a = \sqrt[n]{c} = c^{1/n}$, тобто $m = 1/n$=1/2.5=0.4. Оскільки всі знаки – вірні, то абсолютна похибка числа c не перевищують половини останнього розряду, тобто: Δ_c = 0.05. Тому δ_c=0.05/32.9$\approx$0.0015; $\delta_a = m \cdot \delta_c = 0.4 \cdot 0.0015 = 0.0006$; $a = \sqrt[n]{c} = c^m = 32.9^{0.4} \approx 4.0446$; $\Delta_a = |a| \cdot \delta_a = 4.0446 \cdot 0.0006 \approx 0.0024$. Висновок: число a має три вірних знака, тому $a = 32.9^{0.4} \approx 4.045 \pm 0.002$.

1.2.2. Загальні формули для похибок при обчисленнях функцій

Нехай $u = f(x_1, x_2, ..., x_n)$ – аналітична функція від наближених параметрів. Позначимо $x_i = x_i^o + \Delta x_i$, де x_i^o – номінальні значення параметрів. Розкладемо функцію в околі x_i^o зі збереженням тільки членів наближення першого порядку: $u = f(x_1, x_2, ..., x_n) \approx f(x_1^o, x_2^o, ..., x_n^o) + \sum_{i=1}^{n} \left(\partial f(x_1, x_2, ..., x_n) / \partial x_i \right) \Delta x_i$. Зі вказаною точністю другий член цього виразу визначає абсолютну похибку обчислення функцій. Для визначення відносних похибок зазвичай використовують наближення: $\left(\partial f / \partial x_i \right) / u \approx \partial \left(\ln f \right) / \partial x_i$.

Отже, загальні формули для граничних значень похибок при обчисленнях функцій мають вигляд:

$$\Delta_u \leq \sum_{i=1}^{n} \left| \frac{\partial u}{\partial x_i} \Delta x_i \right|; \quad \delta_u \leq \sum_{i=1}^{n} \left| \frac{\partial (\ln u)}{\partial x_i} \Delta x_i \right| \quad \text{або} \quad \delta_u \leq \Delta_u / |u|. \tag{1.12}$$

Приклад. Об’єм кулі обчислюється за формулою $V = \pi d^3 / 6$. Знайти його значення, якщо прийняти $\pi = 3.14 + 0.0016$ та $d = 3.7 \pm 0.05$.

Номінальне значення об’єму $V = 3.14 \cdot 3.7^3 / 6 \approx 26.5084$ мм^3.

Параметри функції $x_1 = \pi$, $x_2 = d$, причому $\Delta^+_{x_1} = +0.0016$; $\Delta^-_{x_1} = 0$, а $\Delta_{x_2} = 0.05$. Отримаємо вирази для похідних: $\partial V / \partial \pi = d^3/6$; $\partial V / \partial d = \pi d^2/2$. Згідно з першою формулою (1.12): $\left|\Delta^+_V\right| \le (3.7^3/6)\cdot 0.0016 + (3.14\cdot 3.7^2/2)\cdot 0.05 \approx 1.088$ мм^3, а $\left|\Delta^-_V\right| \le (3.7^3/6)\cdot 0 + (3.14\cdot 3.7^2/2)\cdot 0.05 \approx 1.075$, тому $V \approx 26.5 \pm 1.1$ мм^3; $\delta_V = |\Delta_V| / V \le 1.1/26.5 \approx 0.04$, тобто $\delta_V \le 4\%$.

Відповідно до формули (1.6) при $k = 2$: $\delta_V \le 10^{1-2}/2\cdot 2 \approx 0.025$, тобто $|\Delta_V| \le 0.025\cdot 26.5 \approx 0.7$ мм^3. Дійсно, $k = 2$, оскільки при інших значеннях відносна похибка буде мати інший порядок.

Як бачимо, формули (1.6) та (1.12) дають близькі, але не однакові значення похибок. Але формула (1.12) точніше, оскільки враховує реальні похибки кожного з параметрів функції.

Приклад. Модуль Юнга визначається при згині консольної балки прямокутного перетину за формулою $E = 4Pl^3/(bh^3w)$, де P – згинальна сила; l – довжина балки; h та b – висота та ширина перерізу відповідно; w – прогин кінця балки. Розміри балки: $l = 500 \pm 0.5$ мм; $h = 50 \pm 0.1$ мм; $b = 10 \pm 0.1$ мм. Прикладалася сила $P = 400 \pm 1$ Н, яка викликала прогин $w = 0.78 \pm 0.01$ мм.

Номінальне значення: $E = 4\cdot 400\cdot 500^3/(50^3\cdot 10\cdot 0.78) \approx 2.051\cdot 10^5$ МПа.

Отримаємо вирази для похідних: $\partial E/\partial P = 4l^3/(bh^3w)$, $\partial E/\partial l = 12Pl^2/(bh^3w)$, $\partial E/\partial h = -12Pl^3/(bh^4w)$, $\partial E/\partial b = -4Pl^3/(b^2h^3w)$, $\partial E/\partial w = -4Pl^3/(bh^3w^2)$. Згідно з першою формулою (1.12) $\Delta_E \le \dfrac{4l^2}{bh^3w}\left\{l\Delta_P + P\left[3\Delta_l + l\left(\dfrac{3}{h}\Delta_h + \dfrac{1}{b}\Delta_b + \dfrac{1}{w}\Delta_w\right)\right]\right\}$. Обчислимо, що $\Delta_E \le 0.07040\cdot 10^5$ МПа, а $\delta_E \le \Delta_E / E \approx 0.07040\cdot 10^5 / 2.051\cdot 10^5 \approx 0.034$. Отже, модуль Юнга має значення $E \approx (2.05 \pm 0.07)\cdot 10^5$ МПа.

1.2.3. Спосіб границь

Є інший спосіб оцінки похибки при обчисленнях функцій $u = f(x_1, x_2, ..., x_n)$: спосіб границь. Опишемо його у вигляді алгоритму:

а/ виписати значення параметрів у вигляді $\underline{x}_i \le x_i \le \overline{x}_i$, де $\underline{x}_i$ та $\overline{x}_i$ – відомі граничні значення цих параметрів;

б/ визначити характер змін функції $u = f(x_1, x_2, ..., x_n)$ при змінах значень її параметрів. Якщо при збільшенні значення параметра функція:

- зростає, то запис $\underline{x}_i \le x_i \le \overline{x}_i$ не міняється;
- зменшується, то запис $\underline{x}_i \le x_i \le \overline{x}_i$ змінюється на $\overline{x}_i \ge x_i \ge \underline{x}_i$.

в/ обчислити функцію $u = f(x_1, x_2, ..., x_n)$ двічі: u_L – при лівих та u_R – при правих значеннях параметрів функції. Ці значення – крайні: u_L – найменше (обчислене з нестачею), u_R – найбільше (обчислене з надлишком) з можливих, тобто $u_L \le u \le u_R$.

г/ обчислити середнє значення та границі відхилення:

$$u \approx (u_L + u_R)/2 \pm (u_R - u_L)/2 . \qquad (1.13)$$

Примітка 1.1. Якщо x_i створює локальний екстремум функції, то запис типу $\underline{x}_i \le x_i \le \overline{x}_i$ стає неможливим. Тоді замість нього використовують $x_i \le \overline{x}_i$ (локальний мінімум) або $\underline{x}_i \le x_i$ (локальний максимум).

Очевидна перевага цього способу: він не потребує обчислень похідних.

Приклад. Розглянемо завдання, що сформульовано в другому прикладі п.1.2.2. Результати зведено до таблиці 1.4.

Таблиця 1.4. Етапи оцінки похибки обчислення функції способом границь

етап а/	етап б/	етап в/	етап г/
$399 \le P \le 401$	$399 \le P \le 401$	$E_L \approx 1.982 \cdot 10^5$ $E_R \approx 2.123 \cdot 10^5$	$E \approx (2.053 \pm 0.070) \cdot 10^5$
$499.5 \le l \le 500.5$	$499.5 \le l \le 500.5$		
$9.9 \le b \le 10.1$	$10.1 \ge b \ge 9.9$		
$49.9 \le h \le 50.1$	$50.1 \ge h \ge 49.9$		
$0.77 \le w \le 0.79$	$0.79 \ge w \ge 0.77$		

Оскільки параметри h, b та w стоять у знаменнику, то на етапі б/ їхні праві та ліві граничні значення міняємо місцями. Як бачимо, спосіб границь дав практично те саме значення абсолютної похибки, як і формула (1.12). Відносна похибка $\delta_E \le \Delta_E / E \approx 0.070 \cdot 10^5 / 2.053 \cdot 10^5 \approx 0.034$, тобто теж фактично співпадає з раніше отриманою. Остаточно $E \approx (2.05 \pm 0.07) \cdot 10^5$.

Перша значуща цифра в значенні абсолютної похибки перевищує половину розряду (7 більше, ніж 5). Якщо критерієм визначення вірних цифр є *половина* розряду, то у значенні модуля Юнга ми маємо право вважати вірною тільки одну цифру. Тоді відповідно до другої формули (1.6) похибка $\delta_E \le 10^{1-1} / 2 \cdot 2 \approx 0.25$. Вона приблизно у 7.35 разів більше, ніж отримана вище, тобто ніж 0.034. Якщо критерієм визначення вірних цифр є *повний* розряд, то у значенні модуля Юнга ми маємо право вважати вірною дві цифри. Тоді відповідно до другої формули (1.6) похибка $\delta_E \le 10^{1-2} / 2 \cdot 2 \approx 0.025$, що приблизно у 1.36 разів менше, ніж 0.034. Тобто другий варіант визначення вірних цифр дав більш точний результат.

Отже, вираховування властивостей функції, а не окремого числа – результату, дає більш точну оцінку похибки.

1.2.4. Імовірнісні оцінки похибок

Реальні похибки обчислень зазвичай менше, ніж оцінюються наведеними формулами. Це тому, що реальні похибки окремих складових часто мають:

- різні знаки, тому частково взаємно погашаються;
- менші величини, ніж припускаються.

Тому досить часто застосовують так звану *імовірнісну похибку*. Наведемо лише два приклади.

Імовірнісна абсолютна похибка суми N чисел a_i:

$$\Delta^*_\Sigma \le \Delta \cdot \sqrt{N}, \quad \text{де} \quad \Delta \le \max\{\Delta_{a_1}, \Delta_{a_2}, ...\}. \qquad (1.14)$$

Імовірнісна відносна похибка добутку N чисел a_i:

$$\delta_*^* \le \delta \cdot \sqrt{N}, \quad \text{де} \quad \delta \le \max\{\delta_{a_1}, \delta_{a_2}, ...\}. \tag{1.15}$$

Приклад. Оцінимо імовірнісну абсолютну похибку для виразу $a = (a_1 + a_2 + ... + a_N)/N$ при $N \to \infty$, який відповідає формулі для обчислення середнього арифметичного N чисел.

Відповідно до формули (1.14) маємо: $\Delta_a^* \le \Delta \cdot \sqrt{N}/N = \Delta/\sqrt{N}$. Тому отримаємо, що $\lim\limits_{N\to\infty} \Delta_a^* \le \lim\limits_{N\to\infty} \left(\Delta/\sqrt{N}\right) = \Delta \cdot \lim\limits_{N\to\infty} \left(1/\sqrt{N}\right) = 0$. Тобто значення середнього арифметичного наближених чисел є більш точним числом, ніж окремі складові.

Порівняння формул (1.14) з (1.7) та (1.15) з (1.9) показує, що імовірнісна оцінка похибок визначає більшу точність обчислень, ніж звичайна.

1.3. Дійсні числа у двійковій системі ЕОМ. Оцінки похибок при обчисленнях на ЕОМ

1.3.1. Представлення дійсних чисел у двійковій системі ЕОМ

Форма числа із плаваючим розділовим знаком, при обмеженої кількості розрядів, на багато порядків розширяє *діапазон* дійсних чисел, які можна представити. Загальна кількість чисел у діапазоні теж збільшується. Абсолютна точність представлення чисел зменшується, а відносна – збільшується. Є й інші ефекти, зокрема, й негативні. В сучасних ЕОМ зазвичай реалізують форму числа із плаваючим розділовим знаком, згідно з міжнародним стандартом IEEE754 (1985 та 2008 років).

У двійковій системі (більшість ЕОМ) $b = 2$ та $2^{-1} \le M < 1$, тому формула (1.3), а саме $a = \pm M \cdot b^q$, якщо $a \ne 0.0$, набуває нормалізованого вигляду

$$a = \pm.\alpha_1\alpha_2...\alpha_p \cdot 2^q = \pm(\alpha_1/2^1 + \alpha_2/2^2 + ... + \alpha_p/2^p) \cdot 2^q = \pm 2^q \cdot \sum_{k=1}^{p} \alpha_k 2^{-k}, \tag{1.16}$$

де $\alpha_1 = 1$; $\alpha_k = 0$ або 1 при $k > 1$. Якщо при обчисленнях з'являється число, в якому *після* розділового знака стоїть n нулів, то мантиса "зсовується" вліво на n позицій, а значення q (степінь двійки) зменшується теж на n. Якщо *перед* розділовим знаком є n не нулів, то "зсув" проводиться вправо на n позицій, а значення q збільшується на n. Такі дії називають операцією *нормалізації*, а отриманий запис числа – *нормалізованим* (інакше – *денормалізованим*).

Нормалізація – бажана, але не обов'язкова процедура, і не тільки в двійковій системі числення. В ЕОМ – звичайна (крім дуже близьких до нуля чисел).

Згідно з (1.16) вважається, що знак числа не входить до його мантиси.

Приклад. $a = +2^2(1 \cdot 2^{-1} + 0 \cdot 2^{-2} + 1 \cdot 2^{-3}) = 4 \cdot (1/2 + 0/4 + 1/8) = 4 \cdot (5/8) = 2.5$.

На практиці в ЕОМ є виключення із запису (1.16): для від'ємних чисел *після* проведення операції нормалізації числа його мантиса представляється у вигляді *доповнення* до одиниці (оскільки від'ємне число, яке зберігається у пам'яті, може використовуватися багато разів, то є сенс перетворити його у такий варіант представлення, який при використанні потребує меншу кількість

дій). У такому представленні від'ємного числа обов'язково $\alpha_1 = 0$, а всі останні біти мантиси заповнюють значенням, яке дорівнює різниці між одиницею та мантисою числа: $1.0 - M$. Є ще одна "хитрість". Одиницю (або нуль для від'ємних чисел) на першій позиції нормалізованого числа не зберігають, а "визволена" позиція дозволяє додати до збереження мантиси числа ще один біт.

Для збереження q відводять t бітів, причому перший біт – для знака степені (1 – знак "+", 0 – знак "-" а інші $(t-1)$ бітів – для значення степені.

Приклад. При $t = 8$ маємо $2^{8-1} = 128$. Тобто $-128 \le q \le 128$.

Є й інша трактовка значень цих t бітів: для того, щоб число 0 записувалося в ЕОМ одними нулями, для обчислення значення q (степені двійки) використовується "зсув" Δq:

$$q = q_M - \Delta q; \quad 0 \le q_M = 2^t; \quad \Delta q = 2^{t-1}, \tag{1.17}$$

де q_M – машинне значення степені (те, що реально записується у пам'яті ЕОМ); t – кількість бітів для q_M. Позначимо як $L = q_{\min}$ та $H = q_{\max}$.

Приклад. При $t = 8$ маємо $0 \le q_M \le 2^8 = 256$; $\Delta q = 2^{8-1} = 128$; $L = 0 - 128 = -128$; $H = 256 - 128 = 128$. Тобто $-128 \le q \le 128$.

В ЕОМ, згідно з IEEE754, біти для дійсних чисел розташовують у такому порядку (зліва направо):

- знак числа: 0 або 1 ("+" або "-" відповідно);
- t бітів – для знаку степені та значення степені двійки (основи числення);
- p бітів – для значення мантиси.

Для числа з підвищеною точністю всі додаткові біти додаються до мантиси у "хвіст"; а кількість бітів для степені двійки (t) зазвичай збільшується.

Нумерація бітів проводиться з правого (останнього) біта.

Окрім двійкової системі в ЕОМ іноді як основну застосовують шістнадцятирічну систему числення, в якій під кожне шістнадцятирічне число відводиться по 4 біта. Є приклади застосування й інших систем.

Різні мови програмування можуть застосовувати різні формати для запису дійсних чисел, але обчислення з ними проводяться за форматом співпроцесора. Наприклад, у мові програмування C++ дійсні числа записуються, згідно зі стандартом двійкової арифметики IEEE754, в таких основних форматах: одиничної точності (**float** – 32 біта під число) та подвійної точності (**double** – 64 біта під число). Основні характеристики цих форматів зведено в таблицю 1.5.

Важливою характеристикою ЕОМ вважається число ε_M – машинний іпсилон ($1 + \varepsilon > 1$ при $\varepsilon \ge \varepsilon_M$): відносна (та одночасно абсолютна) відстань від одиниці до наступного раціонального числа у напрямку збільшення. Величина ε відповідає формулі (1.1), тобто $a = A \cdot (1 + \varepsilon)$. Для формату **float** значення $\varepsilon_M \approx 1.192\text{e}{-07}$, а для формату **double** значення $\varepsilon_M \approx 2.220\text{e}{-016}$.

Таблиця 1.5. Основні характеристики форматів дійсних чисел у ПЕОМ з процесором типу i386

Формат	Кількість бітів для		Крайні значення q (L та H)	Діапазон значень (порядок значень) $M_0 \div M_\infty$
	степені, t	мантиси, p		
float (binary32)	8	23	-128	e–45
			128	e+38
double (binary64)	11	52	-1024	e–324
			1024	e+308

Примітка 1.2. У Додатку наведено текст програми, яка визначає та виводить на екран монітора відомості про характеристики, наведені в таблиці 1.5, та ε_M.

Таблиця 1.6. Основні характеристики форматів підвищеної точності дійсних чисел у ПЕОМ з процесором типу i586

Формат	Кількість бітів для		Крайні значення q (L та H)	Діапазон значень (порядок значень) $M_0 \div M_\infty$
	степені, t	мантиси, p		
binary128	15	112	-16384	e–4966 ÷ e+4932
			16384	
binary256	19	236	-262144	e–78984 ÷ e+78913
			262144	

Максимальне значення M_∞ називають *машинною нескінченністю*, а найбільш близьке до нуля M_0 – *машинним нулем*. У випадку нормалізації $M_\infty = 1/M_0$. Для обчислення дуже малих чисел, з метою збільшення точності, зроблено відхилення від загального правила, а саме: для малих чисел операція нормалізації не проводиться. Це дозволяє значно зменшити значення M_0 (табл.1.5 й табл.1.6).

Приклад. У випадку застосування формату **float** відповідно до (1.16):

$$M_\infty = +2^{+128} \cdot \sum_{k=1}^{p} (1 \cdot 2^{-k}) = 2^{128} \cdot 0.9(9) \approx 2^{128} \approx 3.4028234663852886\text{e+}38 \quad \text{або}$$

$$M_\infty = +2^{+128} \cdot (1.0 - 2^{-p}) = 2^{128} \cdot 0.9(9) \approx 2^{128} \approx 3.4028234663852886\text{e+}38.$$

Мантиса числа з p двійкових розрядів зберігає $p_{10} \approx p \cdot \lg(2) \approx 0.3\,p$ вірних десяткових розрядів.

Приклад. При $p = 23$ маємо $p_{10} \approx 23 \cdot \lg(2) \approx 6.9 \approx 7$ розрядів, при $p = 52$: $p_{10} \approx 16$, при $p = 112$: $p_{10} \approx 34$, а при $p = 236$: $p_{10} \approx 71$ розрядів.

Обмежена в ЕОМ кількість бітів для представлення числа не дозволяє представити *безперервний* ряд значень дійсних чисел у діапазоні $[-M_\infty, M_\infty]$. Створюється *дискретний* діапазон дійсних *раціональних*, тобто з обмеженою кількістю знаків, дійсних чисел, які до того ж мають *нерівномірний* крок. Відстань між двома суміжними дійсними раціональними числами швидко збільшується при віддаленні від нуля. Наприклад, відстань від нуля до наступного числа дорівнює значенню M_0, а від одиниці до наступного – ε_M, яке вже на багато десяткових порядків більше (див. табл.1.5 та текст під нею).

Виключення. Число +0 містить у всіх бітах нулі. Число -0 має на першій позиції 1, на всіх інших – нулі. Число $-M_\infty$ записується так: перед мантисою всі одиниці, а в мантисі – всі нулі. Число M_∞ відрізняється від $-M_\infty$ нулем на пер-

шій позиції. Є ще "не числа" (No a Numbers, **NaN**), якими позначаються результати неприпустимих операцій та символи. При спробі порівняння завжди **NaN** ≠ **NaN**. Денормалізовані числа, згідно з IEEE754, теж є виключенням.

1.3.2. Виконання арифметичних операцій у ЕОМ

В ЕОМ всі арифметичні операції з дійсними числами виконуються у процесорі дійсних чисел (співпроцесорі). Не вникаючи в деталі, відзначимо, що для виконання арифметичних операцій використовується достатня кількість *додаткових* розрядів (бітів) у регістрах співпроцесора.

Розглянемо два дійсних числа $a = 2^r \cdot m_a \neq 0.0$ та $c = 2^s \cdot m_c \neq 0.0$, які введені до ЕОМ з двійковою системою числення, де m_a та m_c є мантисами цих чисел.

Операції складання/віднімання двох чисел a та c виконуються за виразом:

$$x = a \pm c = \begin{cases} (m_a \pm m_c \cdot 2^{-(r-s)}) \cdot 2^r, & \text{якщо } r > s; \\ (m_a \cdot 2^{-(s-r)} \pm m_c) \cdot 2^s, & \text{якщо } r \le s. \end{cases} \quad (1.18)$$

З урахуванням (1.18), операції складання/віднімання двох чисел мають декілька операцій (етапів), а саме:

- очищення (обнуління) всіх потрібних регістрів співпроцесора;
- знайдення чисел у пам'яті ЕОМ, пересилання до регістрів співпроцесора (операція *вибирання*);
- порівняння значень степенів r та s;
- операція множення майбутнього результату на 2^r або 2^s відповідно: фактично запам'ятовується значення r або s у виділених для цього t бітах;
- виконання операції *вирівнювання порядків*: обчислення $m_c \cdot 2^{-(r-s)}$ або $m_a \cdot 2^{-(s-r)}$. При цьому всі цифри мантиси (нулі та одиниці) зсовуються вправо на $|r-s|$ позицій, на звільнені $|r-s|$ позиції записуються нулі;
- операції складання/віднімання $m_a \pm m_c \cdot 2^{-(r-s)}$ або $m_a \cdot 2^{-(s-r)} \pm m_c$;
- операція нормалізації результату (див. пояснення до формули (1.16));
- операція обрізання результату до стандартної довжини, з осередненням або без нього (є й такий варіант реалізації у ЕОМ) – утворюється x;
- операція запису результату, тобто x, у зовнішню пам'ять ЕОМ.

Операції множення та ділення двох чисел a та c виконується за виразами:

$$x = a \cdot c = (m_a \cdot m_c) \cdot 2^{r+s}; \quad x = a / c = (m_a / m_c) \cdot 2^{r-s}, \quad (1.19)$$

причому для операції ділення число c повинне обов'язково бути нормалізованим. Тут теж є декілька вже описаних вище операцій (етапів).

1.3.3. Оцінки похибок при обчисленнях у ЕОМ

При перевищенні максимального за модулем значення, наприклад, в операціях множення або при спробі ділення на нуль, ЕОМ генерує "логічне переривання" процесу обчислення, яке повинне "оброблятися" алгоритмом. Згідно зі стандартом IEEE754, результат буде призначено як **NaN**. Уникнути такі ситуації часто допомагає зміна порядку розташування чисел або виразів в операціях. Відсутність оброб-

ки переривання програмою може привести до того, що операційна система аварійно закінчить роботу програми.

Як було зазначено у попередньому підрозділі, операція округлення результату до стандартної або потрібної довжини може проводитися з осередненням або без нього. Округлення, згідно з IEEE754, може проводитися "до найближчого" (або до *парного* (12.5 → 12, а не 13; а 13.5 → 14), або до нуля – при виводі на друк або екран), а також "за напрямками": в напрямках нуля, $+\infty$ або $-\infty$.

На різних типах ЕОМ можуть застосовуватися різні правила округлення, їх навіть можна змінювати програмно (це зазвичай використовують компілятори – середовища для створення програм, що можуть виконуватися в ЕОМ). Але максимальна похибка округлення числа A до числа a не може перевищити одиниці останнього збереженого розряду. Тому у двійковій системі та з повним набором бітів, згідно з формулою (1.16):

$$|a - A| = \Delta_a \le 2^{q-p}. \qquad (1.20)$$

Приклад. При $q = 128$ та $p = 23$ (див. табл.1.5) $\Delta_a \le 2^{128-23} \approx 0.40565e+32$. При $q = -128$ та $p = 23$: $\Delta_a \le 2^{-128-23} \approx 0.350325e-45$.

Оскільки мантиса числа a обмежена діапазоном $2^{-1} \le M < 1$, а максимальна похибка округлення в двійковій системі обчислення дорівнює цілому розряду, то гранична відносна похибка округлення числа a може бути знайденою як $\delta_a = \Delta_a / |a| \le 2^{q-p} / 2^q = 2^{-p}$, тобто

$$\delta_a \le 2^{-p}, \qquad (1.21)$$

де p – кількість розрядів для збереження мантиси числа (див. таблиці 1.5 й 1.6).

Машинний іпсилон $\varepsilon_M = 2^{-p}$, тобто співпадає з граничною відносною похибкою округлення чисел у ЕОМ.

Дискретність представлення чисел у ЕОМ не дозволяє виконуватися асоціативному закону при складанні дійсних чисел. Для доказу цього достатньо знайти хоча б один приклад виключення зі цього закону.

Приклад. Обчислити на ПЕОМ суму трьох чисел: $a = 1.0$; $b = 7.35 \cdot 10^{17}$; $c = -7.35 \cdot 10^{17}$, представивши їх як **double**.

Як виявилося, число $d = a + b + c = 0.0$, а число $e = a + (b + c) = 1.0$.

Отже, якщо у сумі з багатьох чисел є два *відносно великих* за модулем числа, які у сумі дають *нуль*, то їх необхідно складати окремо від інших, інакше похибка обчислення може бути дуже великою. Приклад показує, що **асоціативний закон складання в ЕОМ у загальному випадку не виконується**.

Якщо дещо змінити значення b, наприклад, на $b = 7.350001 \cdot 10^{17}$, то результати теж будуть різними: $d = 100000000000.00$, а $e = 100000000001.00$.

Але тут відносна похибка результатів дуже мала.

Якщо ще додатково змінити тип чисел **double** на тип **float**, то результати будуть однакові та дещо несподівані: $d = e = 6.87195 \cdot 10^{10} = b + c$. Тобто число $a = 1.0$ не відображується у результаті, а різниця двох близьких чисел дає підвищену похибку, описану в п.п.1.2.1.4.

Точні операції обчислення є комутативними, асоціативними і дистрибутивними, а на основі арифметики з обмеженою кількістю розрядів – ні. Це необхідно враховувати і при програмуванні алгоритмів.

Втрата точності може виникати при будь-яких математичних операціях. Однак потрібно пам'ятати, що *для фатальної втрати точності всього розв'язку достатньо лише однієї невдалої операції.*

Якщо через $fl(a)$ позначити округлене число a, а через символ $\bullet$ будь-яку просту математичну операцію в ЕОМ (+, –, * або /), то результат такої операції між двох чисел c та d:

$$fl(c \bullet d) = c \bullet d \cdot (1 + \delta), \tag{1.22}$$

причому $d \neq 0$ для операції ділення; відносна похибка δ у загальному випадку є складною функцією c, d, операції, що виконується між ними, характеристики процесора ЕОМ та кількості розрядів, що відводяться для запам'ятовування числа у пам'яті ЕОМ. Але в ЕОМ завжди $\delta \le \varepsilon_M = 2^{-p}$.

Примітка 1.3. Позначення *fl* – від слова *flop*, яке є абревіатурою від *floating-point operation* – операція із плаваючим розділовим знаком.

Будь-яку задачу обчислення можна у загальному розумінні представити у вигляді $\vec{y} = L(\vec{x})$, де L – оператор (правило обчислення), $\vec{x}$ та $\vec{y}$ – вектори з деяких просторів (про векторні простори та оператори див. у Розділі 2). Якщо замість точного $\vec{x}$ використати неточне $\underset{\sim}{\vec{x}} = \vec{x} + \Delta\vec{x}$, то результат зміниться на $\underset{\sim}{\vec{y}} = \vec{y} + \Delta\vec{y}$, тобто отримуємо $\underset{\sim}{\vec{y}} = L(\underset{\sim}{\vec{x}})$. Якщо при $\|\Delta\vec{x}\| \to 0$ гарантовано $\|\Delta\vec{y}\| \to 0$, то алгоритм вважається *стійким*. Якщо $\|\Delta\vec{y}\| \le C \cdot \|\Delta\vec{x}\|$, але число C є великим, то алгоритм має *слабку стійкість* (оператор L має *погану обумовленість*).

Точний аналіз $\|\Delta\vec{y}\|$ може бути дуже складним (джерела похибок описано у підрозділі 1.1). Тому для виявлення похибок, пов'язаних тільки з округленням (п.3 таблиці 1.3), зазвичай аналіз *похибок округлення* пов'язують з аналізом стійкості. Тоді $\vec{x}$ та $\vec{y}$ – вектори без похибок округлення, а $\underset{\sim}{\vec{x}}$ та $\underset{\sim}{\vec{y}}$ – вектори при застосуванні округлення, і похибки округлення можна отримати як $\|\Delta\vec{y}\| = \left\|\underset{\sim}{\vec{y}} - \vec{y}\right\| = \|L(\underset{\sim}{\vec{x}}) - L(\vec{x})\|$, $\delta_y = \|\Delta\vec{y}\| / \|\vec{y}\|$.

Отримаємо вираз для граничної відносної похибки перемножень N дійсних чисел: $y = \prod_{n=1}^{N} x_n$, де всі $x_n \neq 0$. Процес обчислень можна формалізувати як послідовність дій: $y_n = y_{n-1} \cdot x_n$ при $n = 1, 2, ..., N$ та $y_0 = 1.0$. Тоді $y = y_N$. При наявності округлень на кожному кроці маємо інший процес: $\underset{\sim}{y}_n = fl(\underset{\sim}{y}_{n-1} \cdot x_n) = \underset{\sim}{y}_{n-1} \cdot (1.0 + \delta_n) \cdot x_n$, де $|\delta_n| \le \varepsilon_M$, а $\underset{\sim}{y}_0 = 1.0$. Тому

$$\delta_{y_N} = \left|\frac{\underset{\sim}{y}_N - y_N}{y_N}\right| = \frac{\left|\prod_{n=1}^{N}(1.0 + \delta_n) \cdot x_n - \prod_{n=1}^{N} x_n\right|}{\left|\prod_{n=1}^{N} x_n\right|} \le (1.0 + \varepsilon_M)^N - 1.0 = N \cdot \varepsilon_M + O(\varepsilon_M^2). \tag{1.23}$$

Відкидаючи члени вищих порядків малості, остаточно отримаємо, що гранична відносна похибка перемножень N дійсних чисел $\delta_{*_N} \le N \cdot 2^{-p}$.

Отримаємо вираз для граничної відносної похибки суми N дійсних чисел: $y = \sum_{n=1}^{N} x_n$, де всі $x_n > 0$. Процес обчислень можна формалізувати як послідовність дій: $y_n = y_{n-1} + x_n$ при $n = 1,2,...,N$ та $y_0 = 0.0$. Тоді $y = y_N$. При наявності округлень на кожному кроці маємо інший процес: $\underset{\sim}{y}_n = fl(\underset{\sim}{y}_{n-1} + x_n) = (\underset{\sim}{y}_{n-1} + x_n)(1.0 + \delta_n)$, де $|\delta_n| \le \varepsilon_M$, а $\underset{\sim}{y}_0 = 0.0$. Тому

$$\delta_{y_N} = \left|\frac{\underset{\sim}{y}_N - y_N}{y_N}\right| = \left|\sum_{n=1}^{N}(\underset{\sim}{y}_{n-1} + x_n)(1.0+\delta_n) - \sum_{n=1}^{N} x_n\right| \Big/ \left|\sum_{n=1}^{N} x_n\right| \le$$

$$\le \left|(1.0+\varepsilon_M)\cdot\sum_{n=1}^{N}(\underset{\sim}{y}_{n-1} + x_n) - \sum_{n=1}^{N} x_n\right| \Big/ \left|\sum_{n=1}^{N} x_n\right| \le N(N+1)\cdot\varepsilon_M . \qquad (1.24)$$

Тут на останньому кроці опущені дії, викладення яких займає забагато місця [66].

Відкидаючи член вищого порядку малості, остаточно отримаємо, що гранична відносна похибка суми N дійсних чисел має оцінку $\delta_{\Sigma_N} \le N^2 \cdot 2^{-p}$.

Примітка 1.4. Цю формулу доцільно застосовувати лише при великої кількості членів суми, тобто при великих значеннях N.

Приклад. Для обчислення квадрата евклідової норми вектора з N компонентами необхідно виконати N операцій множення та $(N-1)$ складання. Гранична відносна похибка суми буде $\delta_\Sigma \le (N-1)^2 \cdot 2^{-p} \approx N^2 \cdot 2^{-p}$, а множення $\delta_* \le N \cdot 2^{-p}$. Оскільки $\delta_\Sigma >> \delta_*$, приймемо, що загальна гранична відносна похибка $\delta \approx \delta_\Sigma \approx N^2 \cdot 2^{-p}$.

При $p = 23$ (варіант **float**, див. табл.1.5) і при $N = 10^5$ ("рядовий" для крайових задач розмір систем алгебраїчних рівнянь) гранична відносна похибка $\delta \le 10^{10} \cdot 2^{-23} \approx 10^{10} \cdot 1.192 \cdot 10^{-7} \approx 1200$, тобто варіант **float** не можна застосовувати (він дає точність не нижче 0.1% лише при $N \approx 85$ та менше). При $p = 52$ (варіант **double**, див. табл.1.5) і при тому же $N = 10^5$ гранична відносна похибка результату вже значно менша: $\delta \le 10^{10} \cdot 2^{-52} \approx 10^{10} \cdot 2.22 \cdot 10^{-16} \approx 2 \cdot 10^{-6}$.

1.3.4. Практичні рекомендації для зменшення похибок округлення дійсних чисел при обчисленнях на ЕОМ

Для зменшення похибок округлення дійсних чисел при обчисленнях на ЕОМ потрібно:

- зменшувати кількість операцій;
- збільшувати кількість розрядів, тобто застосовувати формати підвищеної точності;
- числа складати у порядку зростання їх абсолютних значень;

- уникати віднімання дуже близьких чисел. Якщо це неможливо, то виконувати ці операції як можна раніше;
- вводити перевірку знаменників (наскільки вони наближені до точного або машинного нуля) або вводити обробку апаратних переривань при діленні на нуль та перевищенні машинної нескінченності;
- групувати обчислення з можливими критичними результатами.

Примітка 1.5. Для ЕОМ сформульовано, на перший погляд, парадоксальний принцип – принцип некомпетентності Пітера: "ЕОМ багаторазово збільшує некомпетентність обчислювача". Тобто користувачем програми, в якій реалізовано деякий алгоритм, може бути малоосвічена людина. І вона буде сприймати будь-які результати обчислень як вірні. Це накладає додаткову відповідальність на фахівців, які розробляють алгоритми та реалізують їх у вигляді програм.

Контрольні питання до підрозділу 1.1

1. Що таке наближене число та його похибка? Які бувають похибки числа та як їх позначають?
2. Наведіть дві класифікації джерел похибок обчислень та причини їх виникнення.
3. Які формати мають позиційна форма запису чисел та чисел із плаваючим розділовим знаком.
4. Для чого введені поняття "значуща цифра" та "вірні знаки"?
5. Як оцінюються похибки наближених чисел, походження яких невідоме?
6. Який зв’язок є між граничною відносною похибкою наближеного числа з кількістю вірних знаків?

Контрольні питання до підрозділу 1.2

1. Як оцінюються похибки при простих обчисленнях з наближеними числами?
2. Як оцінюються похибки при обчисленнях функцій, параметри яких є наближеними числами?

Контрольні питання до підрозділу 1.3

1. Як представляються дійсні числа у двійковій системі ЕОМ?
2. Що таке машинні іпсилон, нуль та нескінченність?
3. За яким алгоритмом проводяться арифметичні операції з дійсними числами в ЕОМ?
4. Яку максимальну відносну похибку може мати будь-яка арифметична операція з "довгими" дійсними числами в ЕОМ?
5. Чому при обчисленнях в ЕОМ не виконується асоціативний закон складання?

Розділ 2

ОСНОВИ ТЕОРІЇ ОПЕРАТОРНИХ РІВНЯНЬ І ФУНКЦІОНАЛЬНОГО АНАЛІЗУ

Розглянемо основні поняття з функціонального аналізу, необхідні для подальшого вивчення чисельних методів та користування спеціальною літературою (припускаючи певні спрощення і фрагментарність).

Всі фізичні поля, що зустрічаються в крайових задачах механіки: температура, переміщення, деформації, напруження й інші, при реалізації в ЕОМ розглядаються у великій кількості точок моделі, яка аналізується. Тому ці поля можна представити у вигляді значень, зібраних у стовпці та матриці. Матриця – це сукупність стовпців. А кожний стовпець можна вважати представником вектора, в якому числа – це компоненти розкладу вектору на базис. Тому теорії лінійних (векторних) просторів, операторних рівнянь і функціонального аналізу є основою для побудови чисельних алгоритмів розв'язування різноманітних крайових задач із застосуванням ЕОМ.

Для скорочення записів застосовують спеціальні позначки, серед них такі: $\forall$ – будь-який; $\equiv$ – рівність за визначенням; $x \in X$ – елемент x належить множині X; $X \subset Y$ – множина X є часткою (включеною до) множини Y; $X \cap Y$ – множина, яка є загальною для множин X і Y (перетинання множин); $X \to Y$ – відображення множини X в множину Y; $\|\bullet\|_X$ – норма у просторі X; R – скалярний простір дійсних чисел; R^n – n-вимірний Евклідів простір; Ω – обмежена область; S або Γ – границя Ω.

2.1. Лінійні (векторні) простори

Простір X з векторів $\vec{u}, \vec{v}, \vec{w}, \ldots$ називається лінійним (або векторним) простором, якщо в ньому припускається наявність двох операцій, а саме *додавання* векторів і *множення* векторів на скаляри з такими властивостями:

1. Для кожної пари векторів $\vec{u}, \vec{v} \in X$ простір X містить і їхню суму $\vec{u} + \vec{v}$, причому

$$\vec{u} + \vec{v} = \vec{v} + \vec{u}; \quad \vec{u} + (\vec{v} + \vec{w}) = (\vec{u} + \vec{v}) + \vec{w}. \tag{2.1}$$

Крім того, простір X містить нульовий вектор $\vec{0}$ і вектор $-\vec{u}$, протилежний вектору $\vec{u}$, такі, що

$$\vec{u} + \vec{0} = \vec{u}; \quad \vec{u} + (-\vec{u}) = \vec{u} - \vec{u} = \vec{0}. \tag{2.2}$$

2. Простір X містить добуток $\alpha\vec{u}$ кожного вектору $\vec{u} \in X$ на будь-який скаляр $\alpha \in R$, причому

$$(\alpha\beta)\vec{u} = \alpha(\beta\vec{u}); \quad 1\vec{u} = \vec{u}; \tag{2.3}$$

$$\alpha(\vec{u} + \vec{v}) = \alpha\vec{u} + \alpha\vec{v}; \quad (\alpha + \beta)\vec{u} = \alpha\vec{u} + \beta\vec{u}, \tag{2.4}$$

де 1 – скалярна одиниця, $\beta \in R$. Якщо α та β є комплексними, то простір теж комплексний лінійний, інакше – дійсний лінійний.

Вираз $\alpha_1\vec{u}_1 + \alpha_2\vec{u}_2 + ... + \alpha_n\vec{u}_n$ називається лінійною комбінацією векторів $\vec{u}_i$.

Кінцева множина з n векторів $\vec{u}_1, \vec{u}_2, ..., \vec{u}_n$ *лінійно незалежна*, якщо із умови $\alpha_1\vec{u}_1 + \alpha_2\vec{u}_2 + ... + \alpha_n\vec{u}_n = \vec{0}$ випливає, що $\alpha_1 = \alpha_2 = ... = \alpha_n = 0$.

Лінійним базисом лінійного простору X є така множина лінійно незалежних векторів $\vec{e}_1, \vec{e}_2, ..., \vec{e}_n$ простору X, що кожний вектор $\vec{u} \in X$ може бути представлений у вигляді лінійної форми $\vec{u} = \alpha_1\vec{e}_1 + \alpha_2\vec{e}_2 + ... + \alpha_n\vec{e}_n$ відносно базисних векторів $\vec{e}_i$. Значення n визначає розмірність базису. Якщо у лінійному просторі для будь-якого цілого числа n можна знайти n незалежних векторів, то він називається *нескінченновимірним* (інакше – *скінченно-вимірним*).

Норма. Лінійний (векторний) простір X називається *нормованим* лінійним (векторним) простором X, якщо для кожного вектора $\vec{u} \in X$ існує таке дійсне число $\| \vec{u} \|_X$ (норма, абсолютна величина, модуль вектора $\vec{u}$), що з $\vec{u} = \vec{v}$ випливає $\| \vec{u} \|_X = \| \vec{v} \|_X$ і що для $\forall \vec{u}, \vec{v} \in X$ та $\forall \alpha \in R$:

$$\| \vec{u} \|_X \geq 0; \quad \text{з } \| \vec{u} \|_X = 0 \text{ випливає, що } \vec{u} = \vec{0}; \quad \| \alpha \vec{u} \|_X = | \alpha | \cdot \| \vec{u} \|_X; \tag{2.5}$$

$$\| \vec{u} + \vec{v} \|_X \leq \| \vec{u} \|_X + \| \vec{v} \|_X \text{ (нерівність Мінковського).} \tag{2.6}$$

Нормований простір X називається *строго нормованим*, якщо $\| \vec{u} + \vec{v} \|_X = \| \vec{u} \|_X + \| \vec{v} \|_X$ можливо лише у випадку $\vec{u} = \lambda \cdot \vec{v}$, де $\lambda > 0$. У строго нормованому просторі X для кожного $\vec{u} \in X$ та кожного підпростору $Y \subset X$ може бути не більше одного найкращого наближення $\vec{u} \in X$ векторами з підпростору Y.

Одиничний вектор – вектор з одиничною нормою.

Дві норми $\| \bullet \|_X$ і $\| \underline{\bullet} \|_X$ називаються еквівалентними, якщо існують два додатних числа α_1 і α_2 таких, що

$$\alpha_1 \| \underline{\vec{u}} \|_X \leq \| \vec{u} \|_X \leq \alpha_2 \| \underline{\vec{u}} \|_X \quad \text{для } \forall \vec{u} \in X. \tag{2.7}$$

У скінченновимірному лінійному просторі всі норми є еквівалентними.

Метрика. Кожний нормований векторний простір є *метричним* простором з метрикою (відстанню) $\rho_{\vec{u},\vec{v}} = \| \vec{u} - \vec{v} \|_X$, в якому визначена *збіжність*. Для метрики простору виконуються майже ті ж властивості, що й для норми: $\rho_{\vec{u},\vec{v}} \geq 0$; з $\rho_{\vec{u},\vec{v}} = 0$ випливає, що $\vec{u} = \vec{v}$; $\rho_{\vec{u},\vec{v}} = \rho_{\vec{v},\vec{u}}$; $\rho_{\vec{u},\vec{v}} \leq \rho_{\vec{u},\vec{w}} + \rho_{\vec{w},\vec{v}}$.

Повнота. Послідовність векторів $\vec{u}_1, \vec{u}_2, ... = \{\vec{u}_n\} \in X$ називається *фундаментальною*, якщо для будь-якого $\varepsilon > 0$ є номер $N(\varepsilon)$ такий, що при будь-яких числах $n > N$ та m виконується нерівність $\| \vec{u}_{n+m} - \vec{u}_n \| < \varepsilon$. Будь-яка послідовність з X, яка збігається, є фундаментальною (зворотне твердження не завжди є вірним).

Вектор $\vec{u}$ називається границею послідовності $\{\vec{u}_n\} \subset X$; $n = 1, 2, ...$, якщо $\lim\limits_{n \to \infty} \| \vec{u}_n - \vec{u} \|_X \to 0$. Спрощена форма запису: $\vec{u} = \lim\limits_{n \to \infty} \vec{u}_n$.

Нормований векторний простір є *повним* в тому і тільки тому випадку, якщо кожна послідовність векторів $\{\vec{u}_n\} \subset X$, яка задовольняє умові

$$\lim_{n\to\infty,\ m\to\infty} \left\|\vec{u}_n - \vec{u}_m\right\|_X = 0 \qquad (2.8)$$

(послідовність Коші), збігається до деякого вектора $\vec{u} \in X$. Інакше кажучи, кожний вектор $\vec{u}_m \in X$, $m \to \infty$, є границею послідовності $\{\vec{u}_n\} \subset X$; $n = 1, 2, \ldots$.

Повний нормований векторний простір називається ***Банаховим*** (Банах Стефан, 1892-1945 рр.).

Кожний *скінченновимірний* нормований векторний простір є повним.

Якщо простір неповний, то його завжди можна вкласти в деякий інший *повний* простір, що називається *поповненням* до неповного.

Приклад. Неповний метричний простір – послідовність {1, 3/4, 13/15,...} сходиться до *ірраціонального* числа $\pi/4$, але не сходиться в просторі *раціональних* чисел. Якщо простір ірраціональних чисел поповнити простором раціональних чисел, то отримаємо повний простір *дійсних* чисел.

Приблизна повнота. *Зліченна* система векторів $\{\vec{u}_n\}$ нормованого простору X називається *повною*, якщо множина лінійних комбінацій цих векторів *всюди щільна* в X, тобто який би не був вектор $\vec{v} \in X$, для кожного $\varepsilon > 0$ знайдуться такі числа $\alpha_1, \alpha_2, \ldots, \alpha_N$ (де N залежить і від $\vec{v}$, і від ε), що

$$\left\|\vec{v} - \sum_{n=1}^{N} \alpha_n \vec{u}_n\right\|_X < \varepsilon. \qquad (2.9)$$

Нормований простір із зліченною повною системою векторів називається *сепарабельним*. Всі повні нормовані простори є сепарабельними.

Інші властивості.

Околом точки, яка визначається вектором $\vec{u}_0$, називається простір $\aleph(\vec{u}_0, r) = \{\vec{u} \in X : \|\vec{u} - \vec{u}_0\|_X < r\}$, який ще називають *відкритою кулею*. Якщо $\aleph(\vec{u}_0, r) = \{\vec{u} \in X : \|\vec{u} - \vec{u}_0\|_X \le r\}$, то цей простір називають *замкненою кулею,* а $\aleph(\vec{u}_0, r) = \{\vec{u} \in X : \|\vec{u} - \vec{u}_0\|_X = r\}$ визначає *сферу*.

Простір $Y \subset X$ називається *відкритим*, якщо кожна його точка має деякий окіл, тобто для $\forall \vec{u}_0 \in Y$ знайдеться $r > 0$ таке, що $\aleph(\vec{u}_0, r) \subset Y$. Інакше – *замкнений* простір. Замкнений простір міститься в деякій замкненій кулі.

Лінійний (векторний) простір $Y \subset X$, де X – нормований простір, також є нормованим з тією же нормою. Якщо простір Y – замкнений, то він називається *підпростором* простору X.

Відстань $\rho_{\vec{u},Y}$ від вектора $\vec{u}$ до підпростору Y визначається виразом $\rho_{\vec{u},Y} = \inf_{\vec{v}\in Y} \|\vec{u} - \vec{v}\|$.

Підпростір Y є *щільним* у X, якщо для $\forall \vec{u} \in X$ та $\forall \varepsilon > 0$ знайдеться вектор $\vec{v} \in Y$ такий, що $\|\vec{u} - \vec{v}\|_X \le \varepsilon$. При цьому говорять про *наближення* простору X векторами з підпростору Y. Якщо є деякий $\vec{v}^* \in Y$ такий, що $\|\vec{u} - \vec{v}^*\|_X = \varepsilon$, то $\vec{v}^*$ називається *найкращим наближенням* $\vec{u} \in X$ векторами з підпростору Y. Вектор $\vec{v}^*$ може не бути єдиним, а також може не існувати. У *скінченновимірному* просторі найкраще наближення є завжди.

Лема Рісса визначає властивість "майже ортогональності" у нормованому векторному просторі: якщо $Y \subset X$ та $Y \neq X$, то для будь-якого $\varepsilon \in [0,1]$ є вектор $\vec{u}_\varepsilon \notin Y$ з нормою $\| \vec{u}_\varepsilon \| = 1$ такий, що $\rho_{\vec{u}_\varepsilon, Y} > 1 - \varepsilon$.

Ермітовий простір. Лінійний (векторний), у загальному випадку комплексний, простір X називається *Ермітовим* (*унітарним*) векторним простором, якщо можна визначити бінарну операцію, яка ставить кожній парі $\vec{u}, \vec{v}$ векторів з X у відповідність скаляр $(\vec{u}, \vec{v})$ – *скалярний добуток* $\vec{u}$ і $\vec{v}$, причому:

1. $(\vec{u}, \vec{v}) = \overline{(\vec{v}, \vec{u})}$ – ермітова симетрія ($\overline{(\vec{v}, \vec{u})}$ комплексно спряжено з $(\vec{u}, \vec{v})$);
2. $(\vec{u}, \vec{v} + \vec{w}) = (\vec{u}, \vec{v}) + (\vec{u}, \vec{w})$ – дистрибутивний закон;
3. $(\vec{u}, \alpha \vec{v}) = \alpha (\vec{u}, \vec{v})$ – асоціативний закон;
4. $(\vec{u}, \vec{u}) \geq 0$; з $(\vec{u}, \vec{u}) = 0$ випливає $\vec{u} = \vec{0}$ – (позитивна визначеність).

Якщо Ермітовий векторний простір є *дійсним*, то всі скалярні добутки – дійсні, і скалярне множення векторів – комутативне: $(\vec{u}, \vec{v}) = (\vec{v}, \vec{u})$.

Нерівність Коші-Буняковського-Шварца:

$$|(\vec{u}, \vec{v})| \leq \| \vec{u} \|_X \| \vec{v} \|_X \quad \text{для } \forall \vec{u}, \vec{v} \in X. \tag{2.10}$$

Будь-який векторний простір зі скалярним добутком може бути нормованим, якщо в ньому ввести норму, наприклад, за формулою $\| \vec{u} \|_X = (\vec{u}, \vec{u})^{1/2}$.

Гілбертів простір. Повний Ермітовий (унітарний) векторний простір називається *Гілбертовим* (Гілберт Давид, 1862-1943 рр.) і зазвичай позначається як H. Всі повні простори векторів, послідовностей і функцій є Гілбертовими. Гілбертові простори одночасно є Банаховими (але не навпаки).

Скінченновимірні дійсні Ермітові (унітарні) векторні простори називаються *Евклідовими*. Вони є повними.

Якщо $(\vec{u}, \vec{v}) = 0$, то вектори $\vec{u}$ та $\vec{v}$ є *ортогональними*, тобто $\vec{u} \perp \vec{v}$. Якщо система векторів $\vec{u}_1, \vec{u}_2, ..., \vec{u}_n$ є ортогональною, то вектори $\vec{u}_1, \vec{u}_2, ..., \vec{u}_n$ є лінійно незалежними. Якщо система векторів $\vec{u}_1, \vec{u}_2, ..., \vec{u}_n$ має властивість $(\vec{u}_m, \vec{v}_n) = \delta_{mn}$, де $m, n = 1, ..., N$, то така система називається *ортонормованою*.

Якщо Гілбертів простір є *сепарабельним*, то в ньому існує ортогональний базис, створений зі скінченної або зліченої кількості векторів. Справедливо і зворотне твердження.

Фундаментальна теорема про проєкції. Для кожного вектору $\vec{u} \in \Omega \subset H$ існує лише один вектор $\vec{u}^* \in \Omega$ такий, що

$$\| \vec{u} - \vec{u}^* \| < \| \vec{u} - \vec{u}^{\#} \|, \tag{2.11}$$

де вектор $\vec{u}^{\#} \in \Omega$ – будь-який інший вектор (не $\vec{u}^*$). Необхідною та достатньою умовою виконання (2.11) є ортогональність вектору $\vec{u} - \vec{u}^*$ будь-якому вектору $\vec{w} \in \Omega$, тобто $(\vec{u} - \vec{u}^*, \vec{w}) = 0$. Важливо, що Ω є *замкненим* підпростором у H.

Простором *Соболева* $W_p^l(\Omega)$ (або $\mathrm{H}^l(\Omega)$) називається *замкнений* векторний простір $\Omega \subset H$ зі скалярним добутком вигляду

$$(\vec{u}, \vec{v})_{W_p^l} = \int_\Omega \sum_{q=0}^{l} \sum_q \frac{\partial^q \vec{u}}{\partial x_1^{k_1} ... \partial x_n^{k_n}} \frac{\partial^q \vec{v}}{\partial x_1^{k_1} ... \partial x_n^{k_n}} d\Omega \tag{2.12}$$

і нормою

$$\|\vec{u}\|_{W_p^l} = (\vec{u},\vec{u})_{W_p^l}^{1/p}, \tag{2.13}$$

де l – максимальний порядок похідних в (2.12); символ $\sum_q$ означає складання по всім похідним порядку q, а $k_1 + ... + k_n = q \geq 0$, причому похідні в (2.12) – безперервні, обмежені, розуміються узагальнено. Зазвичай приймають $p = 2$.

2.2. Лінійні оператори

Нехай X і Y – лінійні нормовані простори, D – деякий підпростір X. Якщо вектору $\vec{u} \in D \subset X$ за заданим правилом зіставлений вектор $\vec{f} = L\vec{u} \in Y$, то в D задано оператор L зі значеннями в Y. Множина D називається областю *визначення* оператора L, позначається як $D(L)$. Множина всіх векторів $\vec{f} = L\vec{u}$ при $\vec{u} \in D(L)$ називається областю *значень* оператора L і позначається як $R(L)$. Інше визначення оператора L: це відображення простору X в простір Y, позначається як $L: X \to Y$. Часто вектор $\vec{f}$ називають образом, а вектор $\vec{u}$ – його прообразом.

Два оператори L і M називаються рівними, якщо для $\forall \vec{u} \in D(L) = D(M)$ виконується умова $L\vec{u} = M\vec{u}$.

Оператор L, визначений в деякому околі $\aleph(\vec{u}_0, r)$, називається *безперервним* у точці, яка визначається вектором $\vec{u}_0$, якщо $L\vec{u} \to L\vec{u}_0$ при $\vec{u} \to \vec{u}_0$.

Оператор L є *лінійним*, якщо він адитивний та однорідний:

$$L(\alpha\vec{u} + \beta\vec{v}) = \alpha L\vec{u} + \beta L\vec{v} \quad \text{для } \forall \vec{u},\vec{v} \in D(L) \text{ та для } \forall \alpha,\beta \in R. \tag{2.14}$$

Лінійний оператор L є *обмеженим*, якщо існує таке число C, що

$$\| L\vec{u} \|_Y \leq C \| \vec{u} \|_X \quad \text{для } \forall \vec{u} \in D(L). \tag{2.15}$$

Якщо X та Y – Банахові простори, а лінійний оператор L є безперервним для вектора $\vec{u} = \vec{0}$, то він є безперервним і для $\forall \vec{u} \in D(L)$. Для лінійних операторів поняття обмеження та безперервності є еквівалентними.

У скінченуовимірному просторі будь-який лінійний оператор є обмеженим і безперервним.

Для будь-яких лінійних безперервних операторів $(L + M)\vec{u} = L\vec{u} + M\vec{u}$ та $(\alpha L)\vec{u} = \alpha L\vec{u}$, тобто оператори $L + M$ та αL теж є лінійні та безперервні.

Будь-які лінійні обмежені оператори, що діють з X в Y (дію позначають як $(X \to Y)$), утворюють лінійний *нормований* простір, який позначають як $\Re(X,Y)$. *Нормою* лінійного обмеженого оператора називається величина

$$\| L \| = \sup_{\vec{u} \neq 0} \frac{\| L\vec{u} \|_Y}{\| \vec{u} \|_X} = \sup_{\|\vec{u}\|=1} \| L\vec{u} \|, \tag{2.16}$$

де sup – точна верхня границя.

Область значень лінійних обмежених операторів Y може співпадати з X, при цьому $(X \to X)$. Тоді:

$$L(M\vec{u}) = (LM)\vec{u}; \quad \| LM \| \leq \| L \| \cdot \| M \|. \tag{2.17}$$

Якщо кожному $\vec{f} \in Y$ відповідає тільки один $\vec{u} \in X$, то існує оператор L^{-1}, *обернений до* L, який має область визначення Y і область значень X:

$$L^{-1}(L\vec{u}) = \vec{u}\,; \quad L(L^{-1}\vec{f}) = \vec{f}\,. \tag{2.18}$$

Якщо L – лінійний, то лінійний і L^{-1}. Щоб L мав обернений оператор L^{-1}, необхідно і достатньо, щоб $L\vec{u} = \vec{0}$ тільки при $\vec{u} = \vec{0}$. Щоб L^{-1} існував і був обмеженим, необхідно і достатньо, щоб існувала така постійна $\lambda > 0$, що

$$\| L\vec{u} \| \geq \lambda \| \vec{u} \| \quad \text{для } \forall \vec{u} \in X\,. \tag{2.19}$$

При цьому $\| L^{-1} \| \leq 1/\lambda$. Лінійний оператор L, який має L^{-1}, є *безперервно* обернений, якщо $R(L) = Y$, а також $L^{-1} \in \Re(X,Y)$, тобто L^{-1} є обмеженим.

2.3. Лінійні обмежені оператори в дійсному просторі Гілберта

Нехай H – дійсний векторний простір Гілберта зі скалярним добутком $(\vec{u},\vec{v})$ і нормою $\| \vec{u} \| = \sqrt{(\vec{u},\vec{u})}$.

Розглянемо обмежені лінійні оператори, задані в H, тобто $D(L) = H$.

Оператор L називається:

- *невід'ємним*, якщо $(L\vec{u},\vec{u}) \geq 0$ для $\forall \vec{u} \in H$; спрощене позначення $L \geq 0$;
- *позитивним*, якщо $(L\vec{u},\vec{u}) > 0$ для $\forall \vec{u} \in H$, крім $\vec{u} = \vec{0}$; спрощене позначення $L > 0$;
- *позитивно визначеним*, якщо $(L\vec{u},\vec{u}) \geq \alpha \| \vec{u} \|^2$ для $\forall \vec{u} \in H$; $\alpha > 0$; (або $L \geq \alpha I$, I – одиничний оператор).

Число $(L\vec{u},\vec{u})$ називається *енергією* невід'ємного оператора L.

Порівняння операторів за *енергією* проводиться за правилом: якщо $((L-M)\vec{u},\vec{u}) \geq 0$ для $\forall \vec{u} \in H$, то оператор $L \geq M$.

Якщо лінійні оператори L та L^* задано на H, і для $\forall \vec{u},\vec{v} \in H$ справедлива *тотожність Лагранжа*

$$(L\vec{u},\vec{v}) = (\vec{u},L^*\vec{v})\,, \tag{2.20}$$

то оператор L^* називається спряженим до оператора L. Якщо лінійний оператор L є обмеженим, то L^* є визначеним *однозначно* і також є *лінійним* і *обмеженим*, має норму $\| L^* \| = \| L \|$ (*це завжди так у скінченновимірному дійсному просторі Гілберта*).

Лінійний обмежений оператор L називається *самоспряженим* (*ермітовим*), якщо $L^* = L$, тобто $(L\vec{u},\vec{v}) = (\vec{u},L\vec{v})$ для $\forall \vec{u},\vec{v} \in H$. Лінійний обмежений оператор L називається *унітарним*, якщо $L^*L = LL^* = I$, тобто $\overline{L^* = L^{-1}}$ та $\| L \| = 1$.

У дійсному просторі Гілберта оператор L^*, спряжений з лінійним оператором L, називається ще *транспонованим* і позначається як L^T.

Оператор L називається:

- *симетричним* (або *самоспряженим*), якщо

$$L^T = L, \text{ тобто } (\vec{u},L\vec{v}) = (\vec{v},L\vec{u}) \text{ для } \forall \vec{u},\vec{v} \in H\,; \tag{2.21}$$

- *косо*(*анти*)*симетричним*, якщо

$$L^T = -L \text{, тобто } (\vec{u}, L\vec{v}) = -(\vec{v}, L\vec{u}) \text{ для } \forall \vec{u}, \vec{v} \in H; \quad (2.22)$$

- *ортогональним*, якщо

$$L^T L = LL^T = I \text{, тобто } L^T = L^{-1} \text{ та } \| L \| = 1. \quad (2.23)$$

Добуток ортогональних операторів є ортогональним оператором. Добуток двох симетричних операторів L та M є симетричним оператором, якщо $LM = ML$. Сума двох симетричних операторів є симетричним оператором.

Для будь-якого позитивного самоспряженого оператора в дійсному просторі Гілберта справедлива узагальнена нерівність Коші-Буняковського

$$(L\vec{u}, \vec{v})^2 \le (L\vec{u}, \vec{u})(L\vec{v}, \vec{v}). \quad (2.24)$$

Оператор M називається *квадратним коренем* з оператора L, якщо $M^2 = L$; позначається як $L^{1/2}$.

2.4. Лінійні обмежені функціонали в дійсному просторі Гілберта

Будь-який оператор L з векторними параметрами, *значеннями* якого є *дійсні або комплексні числа*, називається *функціоналом.* Зазвичай позначається як $J(\vec{u})$. Зазвичай функціоналами вважають оператори L у вигляді визначеного інтегралу.

Функціонал є *лінійним*, якщо $J(\alpha\vec{u} + \beta\vec{v}) = \alpha J(\vec{u}) + \beta J(\vec{v})$ для $\forall \vec{u}, \vec{v} \in H$ та $\alpha, \beta \in R$. Функціонал є *обмеженим*, якщо для $\forall \vec{u}, \vec{v} \in H$ є обмеженою величина $\| J(\vec{u}) \| = \sup\limits_{\vec{u} \in D(J),\ \|\vec{u}\| \le 1} |(\vec{u}, \vec{v})|$.

Кожний *лінійний*, обмежений у просторі Гілберта функціонал $J(\vec{u})$ можна представити у вигляді скалярного добутку $J(\vec{u}) = (\vec{u}, \vec{v})$, де $\vec{v} \in H$ – деякий *єдиний* вектор (*теорема Ф. Рісса* (угор. Fr. Riesz), 1880-1956 рр.).

Нехай $H \otimes H$ – прямий добуток простору Гілберта самого на себе. Функціонал $F(\vec{u}, \vec{v})$, заданий на $H \otimes H$, називається *білінійним* (*білінійною формою*), якщо він є лінійним окремо по кожному з аргументів.

Якщо $\vec{u} = \vec{v}$, то функціонал $F(\vec{u}, \vec{u})$ можна вважати заданим на H. Такий функціонал $F(\vec{u}, \vec{u})$ називається *квадратичним функціоналом на* H.

У чисельних методах для побудови алгоритмів розв'язуванні лінійних крайових задач основою є наступні теореми.

Теорема 1 (єдиного розв'язку). Якщо оператор L в рівнянні $L\vec{u} = \vec{f}$ позитивно визначений, то це рівняння має не більше одного розв'язку.

Теорема 2. Якщо в просторі Гілберта рівняння $L\vec{u} = \vec{f}$, де L – позитивно визначений симетричний лінійний оператор, має розв'язок $\vec{u}_0 \in X$, то він мінімізує функціонал

$$F(\vec{u}) = \frac{1}{2}(L\vec{u}, \vec{u}) - (\vec{u}, \vec{f}). \quad (2.25)$$

Справедливо і зворотне твердження.

Теорема 3 (існування). Нехай $F(\vec{u})$ – функціонал вигляду (2.25). Вектор $\vec{u}$ з обмеженою енергією, який мінімізує $F(\vec{u})$, існує тоді і тільки тоді, коли оператор L позитивно обмежений знизу.

Множина $Z \subset H$ називається *опуклою*, якщо із умови $\vec{u},\vec{v} \in Z$ випливає, що

$$(1-\omega)\vec{u} + \omega\vec{v} \in Z \quad \text{для } \forall \omega \in [0,1]. \tag{2.26}$$

Функціонал $F(\vec{u})$, заданий на H, називається *опуклим*, якщо

$$F((1-\omega)\vec{u} + \omega\vec{v}) \le F(\vec{u}) + \omega(F(\vec{v}) - F(\vec{u})) \quad \text{для } \forall \omega \in [0,1]. \tag{2.27}$$

З (2.27) випливає, що функціонал типу (2.25) є опуклим, якщо оператор L є позитивно визначеним.

Використовуючи поняття (властивість) опуклості (2.27) функціоналів, доводиться **варіаційна теорема**: якщо в просторі Гілберта задано операторне рівняння $L\vec{u} = \vec{f}$, яке має розв'язок $\vec{u}_0$ (L – позитивно визначений симетричний лінійний оператор), а також існує вектор $\vec{v}_0 \in \Omega$, який реалізує мінімум функціонала $F(\vec{v}) = (L\vec{v},\vec{v}) - 2(\vec{v},\vec{f})$ на замкнутій опуклій множині $\Omega \subset H$, то для цього вектору існує білінійна форма, яка відповідає нерівності

$$a(\vec{u}_0, \vec{v} - \vec{v}_0) \ge (\vec{f}, \vec{v} - \vec{v}_0) \quad \text{для } \forall \vec{v} \in \Omega. \tag{2.28}$$

Якщо ж $\Omega \equiv X$, то замість (2.28)

$$a(\vec{u}_0, \vec{v}) = (\vec{f}, \vec{v}) \quad \text{для } \forall \vec{v} \in X. \tag{2.29}$$

Нерівності вигляду (2.28) називаються варіаційними, а формула (2.29) – варіаційним рівнянням Ейлера. Воно є умовою стаціонарності функції багатьох змінних, що виражається в рівності нулю її частинних похідних.

Варіацією δy функції $y(x)$ називається *різниця* $\delta y(x) = Y(x) - y(x)$, де $Y(x)$ – близька за значеннями до $y(x)$ *інша* функція з тими же обмеженнями, що й $y(x)$. Варіаційний аналіз – це теорія визначення змін значень та знаходження екстремумів визначеного інтеграла (від функцій, що варіюються).

2.5. Нелінійні обмежені функціонали в просторі Гілберта

Щоб отримувати частинні похідні у векторному просторі, потрібно ввести похідні та диференціали векторного простору. Проблема в тому, що не можна мати у знаменнику вектор. Є два варіанта диференціалів та похідних в H.

Нехай X та Y – два Гілбертових дійсних простори, а L – повний в X нелінійний оператор, що діє з $D(L) \subset X$ в Y.

Якщо для деякого $\vec{u} \in X$ при будь-якому $\vec{v} \in X$

$$\lim_{\alpha \to 0} \left\| \frac{1}{\alpha} \left[L(\vec{u} + \alpha\vec{v}) - L(\vec{u}) \right] - GL(\vec{u},\vec{v}) \right\| = 0, \tag{2.31}$$

де $\alpha \in R$, то $GL(\vec{u},\vec{v})$ – *диференціал Гато* (Рене Ежен Гато, 1860-1914 рр.) оператора L в точці, що визначається вектором $\vec{u}$, при прирощенні $\alpha\vec{v}$. Крім того, $GL(\vec{u},\vec{v})$ є однорідним по $\vec{v}$ та не обов'язково лінійним по $\vec{u}$. *Похідна Гато* $D_G L(\vec{u})$ оператора L в точці, що визначається вектором $\vec{u}$, вводиться як лінійний оператор, що діє на $\vec{u}$:

$$GL(\vec{u},\vec{v}) = D_G L(\vec{u})\vec{v}\,. \tag{2.32}$$

Якщо для деякого $\vec{u} \in X$ при будь-якому $\vec{v} \in X$

$$L(\vec{u}+\vec{v}) - L(\vec{u}) = \delta L(\vec{u},\vec{v}) + \omega(\vec{u},\vec{v}),$$

де $\delta L(\vec{u},\vec{v})$ є лінійним оператором відносно $\vec{v}$, а також

$$\lim_{\|\vec{v}\|\to 0} \frac{1}{\|\vec{v}\|} \left\|\omega(\vec{u},\vec{v})\right\| = 0, \tag{2.33}$$

то $\delta L(\vec{u},\vec{v})$ – *диференціал Фреше* (Моріс Рене Фреше, 1878-1973 рр.) оператора L в точці, що визначається вектором $\vec{u}$, при прирості $\vec{v}$, а $\omega(\vec{u},\vec{v})$ – залишковий член (залишок).

Внаслідок лінійності $\delta L(\vec{u},\vec{v})$ відносно $\vec{v}$, оператор $L'(\vec{u})$, що діє на $\vec{v}$, зветься *похідною Фреше* оператора L в точці, яка визначається вектором $\vec{u}$:

$$\delta L(\vec{u},\vec{v}) = L'(\vec{u})\vec{v}\,. \tag{2.34}$$

Якщо існує диференціал Фреше $\delta L(\vec{u},\vec{v})$, то існує і *рівний йому* диференціал Гато $GL(\vec{u},\vec{v})$ (зворотне – тільки для безперервних у точці $\vec{u}$ похідних).

Оператор R, що визначається формулою:

$$(R(\vec{u}),\vec{v}) = \lim_{\alpha\to 0} \frac{1}{\alpha}\left[P(\vec{u}+\alpha\vec{v}) - P(\vec{u})\right], \tag{2.35}$$

де $P(\vec{u})$ – функціонал, який диференціюється за Фреше (тому й за Гато), називається *градієнтом* функціонала $P(\vec{u})$, тобто $R(\vec{u}) = grad\, P(\vec{u})$, а $R(\vec{u})$ є вектором.

Якщо $grad\, P(\vec{u}) = R(\vec{u})$ для $\forall \vec{u} \in \Omega \subset X$, то оператор R називається *потенційним* оператором. *Симетричний* оператор є потенційним оператором. Є й інші варіанти потенційних операторів (див. параграф 23 книги [7]).

Точка, що визначається вектором $\vec{u}_0 \in X$, така, що $P(\vec{u}) \le P(\vec{u}_0)$ або $P(\vec{u}) \ge P(\vec{u}_0)$ для всіх $\vec{u} \in \aleph(\vec{u}_0, r)$, називається *екстремальною точкою* функціонала $P(\vec{u})$. Тут $\aleph(\vec{u}_0, r)$ – окіл точки, яка визначається вектором $\vec{u}_0$.

Теорема 4 о критерії потенційності оператора (М.М. Вайнберг, 1956 р.). Нехай R – оператор, *потенційний* в $\aleph(\vec{u}_0, r)$. Тоді існує функціонал $P(\vec{u})$, єдиний з точністю до константи, градієнтом якого є R:

$$P(\vec{u}) = P(\vec{u}_0) + \int_0^1 \left(R(\vec{u}_0 + s(\vec{u}-\vec{u}_0)), \vec{u}-\vec{u}_0\right) ds\,, \tag{2.36}$$

де $\vec{u}_0 \in X$ є довільним вектором. Часто обирають $\vec{u}_0 = \vec{0}$; $P(\vec{u}_0) = 0$.

Ця теорема є узагальненням **теореми 2** на нелінійний випадок, дозволяє знаходити екстремуми нелінійного функціонала $P(\vec{u})$ із умови

$$(grad\, P(\vec{u}_0), \vec{u}) = 0 \tag{2.37}$$

для екстремальної точки та для будь-якого вектору $\vec{u} \in H$.

Наведемо пояснення до теореми 4. Екстремальній точці $\vec{u}_0$ функціонала $P(\vec{u})$ відповідає його екстремальне значення $P(\vec{u}_0)$ (див. рис.2.1, двомірне зображення). Вектор $\vec{u}-\vec{u}_0$ дає поточну відстань від екстремальної точки функціонала $P(\vec{u})$. Вектор $\vec{u}_0 + s(\vec{u}-\vec{u}_0)$ визначає точку в околі екстремальної

точки $\vec{u}_0$, причому при збільшенні s від 0 до 1 (границі інтегрування) ця точка переміщується з точки, що визначається $\vec{u}_0$, до точки, що визначається поточним вектором $\vec{u}$, тобто на максимальну відстань $r = \| \vec{u} - \vec{u}_0 \|$. Вектор $R(\vec{u}) = grad\, P(\vec{u})$ зображується як нормаль до поверхні $P(\vec{u})$ в точці, що визначається поточним вектором $\vec{u}$.

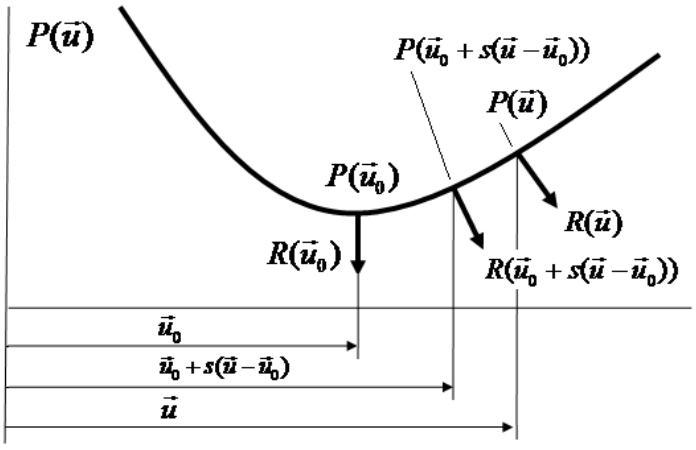


Рис.2.1. До пояснення теореми 4

Якщо для вектору $R(\vec{u}_0 + s(\vec{u} - \vec{u}_0))$ величина $s = 0$, то $R(\vec{u}_0)$ має вертикальний напрямок, створює *прямий* кут з вектором $\vec{u} - \vec{u}_0$, тому скалярний добуток $\left(R(\vec{u}_0), \vec{u} - \vec{u}_0\right) = 0$. При $s = 1$ маємо вектор $R(\vec{u})$, який, при наявності в околі $\aleph(\vec{u}_0, r)$, де величина $r = \| \vec{u} - \vec{u}_0 \|$, *єдиної* екстремальної точки $\vec{u}_0$, вже не є вертикально направленим (див. рис.2.1), створює з вектором $\vec{u} - \vec{u}_0$ *не прямий* кут, тому скалярний добуток $\left(R(\vec{u}), \vec{u} - \vec{u}_0\right)$ вже відмінний від нуля, тобто щось додає до $P(\vec{u}_0)$ або віднімає від $P(\vec{u}_0)$: в точці, що визначається поточним вектором $\vec{u}$, вже немає екстремуму функціонала $P(\vec{u})$. Оскільки $R(\vec{u}) = grad\, P(\vec{u})$, то $P(\vec{u})$ можна обчислити через $R(\vec{u})$ операцією інтегрування у всьому околі, тому границі інтегрування визначає $s \in [0,1]$. Отже, інтеграл $\int_0^1 \left(R(\vec{u}_0 + s(\vec{u} - \vec{u}_0)), \vec{u} - \vec{u}_0\right) ds$ визначає повну величину зміни значення функціонала $P(\vec{u})$ від екстремального значення до поточного. Якщо така зміна не дорівнює нулю при будь-якої відстані $r > 0$, то в околі $\aleph(\vec{u}_0, r)$ екстремальна точка дійсно є, а оператор R зветься *потенційним* у цьому околі.

2.6. Про наближений розв'язок операторних рівнянь

2.6.1. Про наближений розв'язок лінійних операторних рівнянь

Лінійні задачі механіки твердого тіла, що деформується, можуть бути представлені у вигляді варіаційного рівняння (див. (2.29)):

$$a(\vec{u}, \vec{v}) = (\vec{f}, \vec{v}) \quad \text{для } \forall \vec{v} \in X, \tag{2.38}$$

де $a(\vec{u}, \vec{v})$ – позитивно визначена і безперервна білінійна форма; $(\vec{f}, \vec{v})$ – лінійна безперервна форма.

Нехай $\{h\}$ – послідовність елементів з R, що прямує до нуля, і нехай для $\forall h \in \{h\}$ існує скінченновимірний підпростір $X_h \subset X$.

Теорема 5 (лема Сеа). Якщо позитивно визначена і безперервна білінійна форма має властивості:

$$a(\vec{u},\vec{u}) \geq \alpha \|\vec{u}\|^2 \quad \text{для } \forall \vec{u} \in X,\ \alpha = const > 0;$$

$$a(\vec{u},\vec{v}) \leq \beta \|\vec{u}\| \|\vec{v}\| \quad \text{для } \forall \vec{u}, \vec{v} \in X,\ \ \beta = const > 0,$$

то наближений розв'язок $\vec{u}_h$ задачі

$$a(\vec{u}_h,\vec{v}_h) = (\vec{f}_h,\vec{v}_h) \quad \text{для } \forall \vec{v}_h \in X_h, \tag{2.39}$$

яка є сітковим аналогом задачі (2.38), існує, єдиний і задовольняє умові

$$\|s_h\vec{u}_0 - \vec{u}_h\| \leq \frac{\beta}{\alpha} \inf \|s_h\vec{u}_0 - \vec{v}_h\| \quad \text{для } \forall \vec{v}_h \in X_h, \tag{2.40}$$

де $\vec{u}_0$ – точний розв'язок задачі (2.38), inf – точна нижня границя, s_h – оператор відображення на сітку. Якщо білінійна форма ще і симетрична, то замість (2.40)

$$\|s_h\vec{u}_0 - \vec{u}_h\| \leq \sqrt{\frac{\beta}{\alpha}} \inf \|s_h\vec{u}_0 - \vec{v}_h\| \quad \text{для } \forall \vec{v}_h \in X_h. \tag{2.41}$$

Теорема 6. Якщо білінійна форма $a(\vec{u},\vec{v})$ має такі ж властивості, як і у лемі Сеа, існує простір $W \subset X$, щільний в X, та

$$\lim_{h\to 0} \|s_h\vec{v} - \vec{v}_h\| = 0 \quad \text{для } \forall \vec{v} \in W, \tag{2.42}$$

де s_h – оператор відображення з W в X_h, то

$$\lim_{h\to 0} \|s_h\vec{u}_0 - \vec{u}_h\| = 0, \tag{2.43}$$

тобто *послідовність* наближених розв'язків задачі $\{\vec{u}_h\}$ сходиться до точного $\vec{u}_0$.

2.6.2. Про наближений розв'язок нелінійних операторних рівнянь

Для багатьох нелінійних операторних рівнянь сформульовано умови існування і одиничності розв'язку, які часто є аналогічними умовам, що сформульовано для лінійних операторів.

Зокрема, застосовують теорему 4 підрозділу 2.5. Якщо у нелінійному рівнянні крайової задачі

$$L(\vec{u}) = \vec{f} \tag{2.44}$$

оператор L є *потенційним*, то і у рівнянні $R(\vec{u}) = L(\vec{u}) - \vec{f}$ оператор R теж є потенційним. А *потенційні* нелінійні оператори мають наступну важливу властивість (див. теорему 4): якщо для (2.44) існує розв'язок $\vec{u}_0 \in X$, який є критичною точкою для функціонала $P(\vec{u})$, градієнтом якого є $R(\vec{u})$, то

$$(R(\vec{u}_0),\vec{w}) = 0, \text{ інакше } \quad ((L(\vec{u}_0) - \vec{f}),\vec{w}) = 0 \tag{2.45}$$

для довільного $\vec{w} \in X$. Тобто існує ортогональність векторів $L(\vec{u}_0) - \vec{f}$ та $\vec{w} \in X$. Ця властивість є основою одного з методів побудови алгоритмів знаходження розв'язків у нелінійних задачах (див. Розділ 17).

Ще існує *фундаментальна теорема про проєкції* (див. Розділ 2.1 та формулу (2.11)). Ця теорема теж лежить в основі ще одного загального методу знаходження розв'язків у нелінійних задачах.

Також, як і для лінійних задач, вводиться $\{h\}$ – послідовність елементів з простору R, що прямує до нуля, де для $\forall h \in \{h\}$ існує скінченновимірний підпростір $X_h \subset X$. Для деяких нелінійних задач доведено теореми про те, що при визначених умовах наближений розв'язок $\vec{u}_h$ відповідної нелінійної задачі, яка є сітковим аналогом початкової задачі, існує, єдиний і задовольняє визначеним умовам. У результаті алгебраїзації нелінійних задач одержують нелінійні системи алгебраїчних рівнянь (САР). Для розв'язування таких САР потрібні відповідні методи.

Більш докладний виклад проблем, порушених у цьому Розділі, особливо для нелінійних випадків, вимагає значного місця, більш точних формулювань і спеціальної математичної підготовки. Можна рекомендувати (не повний) список із книг [7, 8, 26, 27, 36, 42, 70].

Контрольні питання до підрозділу 2.1

1. Які основні властивості мають вектори в лінійному просторі?
2. Що таке норма й метрика векторного простору та скалярний добуток в ньому, які їхні властивості? Що таке повнота простору?
3. Наведіть визначення Банахового, Ермітового, Гілбертового й Евклідового просторів та простору Соболєва. Як вони пов'язані між собою?
4. Про що каже фундаментальна теорема про проєкції?

Контрольні питання до підрозділу 2.2

1. Що таке оператор? Які властивості має лінійний оператор?
2. Як визначаються властивість обмеженості та норма оператора?

Контрольні питання до підрозділу 2.3

1. Які додаткові властивості має оператор в Гілбертовому просторі?
2. Що таке енергія оператора?
3. Як визначаються спряжений та самоспряжений оператори? Які властивості мають самоспряжені, кососиметричні та ортогональні оператори?

Контрольні питання до підрозділу 2.4

1. Що таке обмежений функціонал в дійсному Гілбертовому просторі і як він пов'язаний зі скалярним добутком?
2. Які теореми є теоретичною основою для побудови чисельних алгоритмів з розв'язування лінійних крайових задач?

Контрольні питання до підрозділу 2.5

1. Які властивості мають диференціали Гато та Фреше?
2. Про що йдеться в теоремі Вайнберга?

Контрольні питання до підрозділу 2.6

1. Яку оцінку дає лема Сеа? Як впливає симетричність білінійної форми на норму похибки наближеного на сітці розв'язку?
2. На яких теоретичних основах будуються алгоритми знаходження розв'язків нелінійних крайових задач?

Частина II
ЧИСЕЛЬНІ МЕТОДИ АЛГЕБРИ

Розділ 3

НАБЛИЖЕНЕ РОЗВ'ЯЗУВАННЯ ТРАНСЦЕНДЕНТНИХ І АЛГЕБРАЇЧНИХ РІВНЯНЬ

Під розв'язуванням рівняння

$$f(z) = 0 \tag{3.1}$$

мають на увазі знаходження всіх коренів рівняння. *Корені рівняння* (3.1) – це такі значення аргументу z, при яких значення $f(z)$ дорівнює нулю.

Примітка 3.1. Оскільки корені можуть бути комплексними, традиційно аргумент рівняння (3.1) позначають символом z.

Корені можуть бути:

- відособленими або здвоєними (парними);
- дійсними або комплексними (комплексні – завжди парні).

Розв'язування рівняння (3.1) проводиться двома етапами:

- *відділення коренів*: знаходження інтервалів, на яких знаходиться або один відособлений корінь, або тільки один комплект здвоєних коренів;
- почергове *знаходження коренів*.

Алгебраїчні рівняння (на основі степеневих рядів), на відміну від рівнянь загального вигляду (трансцендентних), мають багато специфічних властивостей, тому для них додатково розроблені спеціальні методи відділення коренів та їх знаходження (див. підрозділ 3.2).

3.1. Знаходження коренів трансцендентних рівнянь

3.1.1. Відділення коренів трансцендентних рівнянь

Досі немає універсального методу відділення коренів. Для розв'язання цієї задачі рекомендують використовувати такі властивості функцій:

- якщо безперервна функція $f(z)$ на кінцях інтервалу $[a, b]$ має різні знаки, тобто добуток $f(a) \cdot f(b) < 0$, то в інтервалі $[a, b]$ є принаймні один корінь рівняння (3.1). Це твердження відомо як теорема Коші. Графічні пояснення цієї властивості зображено на рис.3.1. З них можемо зробити висновки про те, що в обраному інтервалі може бути багато коренів (див. рис.3.1-б), а також про те, що корені можуть знаходитися в ньому навіть тоді, коли $f(a) \cdot f(b) > 0$ (див. рис.3.1-г);

- в інтервалі $[a, b]$ корінь може бути тільки один, якщо $f'(z) < 0$ або $f'(z) > 0$ скрізь в $[a, b]$ (див. рис.3.1-а);
- кратні корені функції $f(z)$ одночасно є коренями функції $f'(z) = 0$.

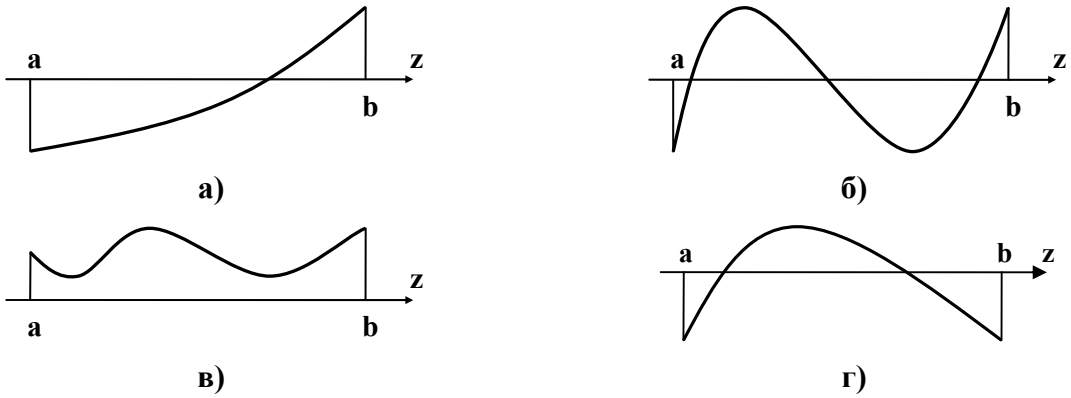


Рис.3.1. Про наявність коренів безперервної функції в інтервалі $[a, b]$

Процес відділення коренів рекомендують проводити таким чином:

- призначити деякий інтервал $[a, b]$, зокрема необмежений $(-\infty,\infty)$, частково обмежений ($(-\infty,b]$ або $[a,\infty)$);
- по можливості максимально повно виявити характер поведінки функції $f(z)$ в призначеному інтервалі. Зокрема, знайти всі екстремальні точки та точки перегину функції. Умови для визначення таких точок відомі: перша похідна $f'(z) = 0$ для екстремальних, друга похідна $f''(z) = 0$ – для точок перегину;
- визначити знаки функції на кінцях інтервалу, а також у деякій кількості проміжних точок, які призначати:
  - з урахуванням виявлених особливостей функції;
  - застосуванням методу половинного ділення інтервалу. Зокрема, для функції, що зображена на рис.3.1-г, цей метод вже при першому діленні дасть два інтервали, на яких є принаймні по одному кореню;
  - застосуванням графічного методу.
- виявити один або декілька інтервалів, що містять корені.

3.1.2. Знаходження коренів трансцендентних рівнянь

Другий етап – власне знаходження коренів. Для знаходження коренів трансцендентних рівнянь існує декілька методів: графічний та ітераційні. Іноді для прискорення збіжності ітераційні методи комбінують.

Майже усі ітераційні методи будуються на представленні формули (3.1) у вигляді

$$z = \vartheta(z) \quad \text{або} \quad z = z - \lambda \cdot f(z), \tag{3.2}$$

де множник λ може бути будь-яким (але обмеженим), оскільки для коренів повинно бути $f(z) = 0$. Очевидно, що у другому рівнянні можемо позначити $z - \lambda \cdot f(z) = \vartheta(z)$.

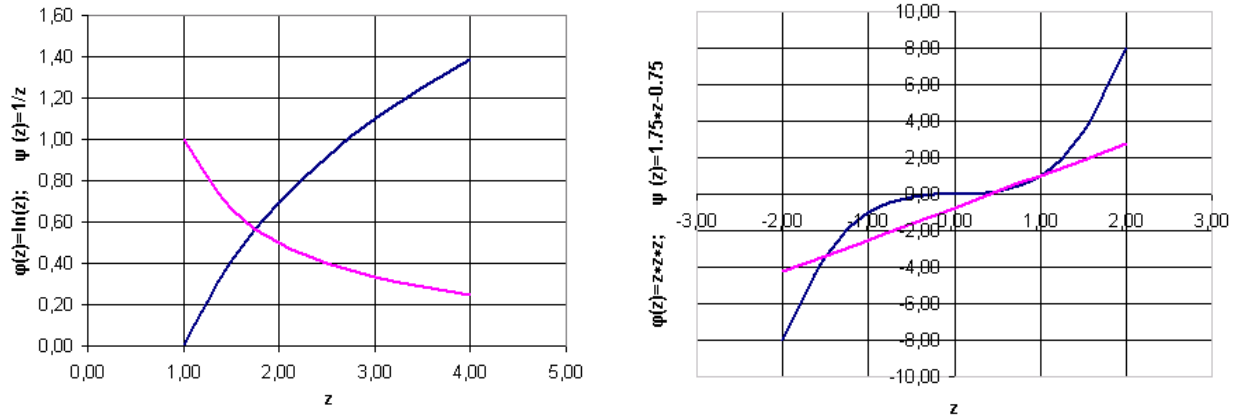
Розглянемо основні методи.

3.1.2.1. Графічний метод

Застосовується для відділення коренів та знаходження коренів із незначною точністю. Для цього будується графік функції $f(z)$, або графіки двох функцій, отриманих шляхом перетворення функції $f(z)$ до виразу $\varphi(z)=\psi(z)$.

Приклад 3.1. Функцію $f(z)=z\cdot\ln(z)-1=0$ перетворимо до вигляду $\ln(z)=1/z$, потім позначимо $\varphi(z)=\ln(z)$ та $\psi(z)=1/z$. Графіки цих функцій зображено на рис.3.2-а. З них робимо висновок, що функція $f(z)=z\cdot\ln(z)-1=0$ має лише один корінь, розташований в інтервалі [1.5, 2].

Приклад 3.2. Функцію $f(z)=z^3-1.75\cdot z+0.75=0$ перетворимо до $z^3=1.75\cdot z-0.75$, потім позначимо $\varphi(z)=z^3$, а $\psi(z)=1.75\cdot z-0.75$. Графіки цих функцій зображено на рис.3.2-б. З них робимо висновок, що функція $f(z)=z^3-1.75\cdot z+0.75=0$ має три кореня, розташованих в інтервалі [-1.6, 1.1].



а) **б)**

Рис.3.2. Графічний метод відділення коренів та їхнє знаходження

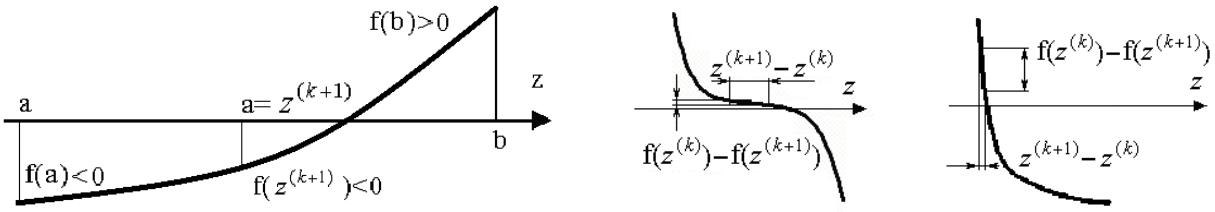
Оскільки за допомогою багатьох програм (Excel, MathCAD, Matlab, Scilab, Mathematica, Maple тощо) можна переглянути графіки практично будь-яких функцій у будь-якому діапазоні, то графічний метод відділення коренів набув значної популярності у цій процедурі.

3.1.2.2. Метод проб (метод ділення навпіл)

В ітераціях поточний інтервал діліться навпіл точкою $z^{(k+1)}$, визначається знак функції $f(z^{(k+1)})$, відкидається та частина поточного інтервалу $[a,b]$, на кінці якої функція мала такий же знак (див. рис.3.3-а). Математично: $z^{(k+1)}=(a+b)/2$; якщо $f(z^{(k+1)})\cdot f(a)>0$, то $a=z^{(k+1)}$, інакше $b=z^{(k+1)}$. Процес повторюється до моменту досягнення призначеної точності. Потрібно використовувати одночасно два критерії досягнення збіжності (див. рис.3.3-б та рис.3.3-в):

$$|f(z^{(k+1)})-f(z^{(k)})|<\varepsilon_f\cdot|f(z^{(k)})| \quad \text{та} \quad |z^{(k+1)}-z^{(k)}|<\varepsilon_z\cdot|z^{(k)}|, \qquad (3.3)$$

де $\varepsilon_f,\varepsilon_z$ – встановлена точність знаходження кореня рівняння (3.1) за значенням функції та аргументу відповідно. Дійсно, виконання тільки першої з умов (3.3) для випадку рис.3.3-б та, навпаки, тільки другої з умов (3.3) для випадку рис.3.3-в, призведе до малої точності знаходження кореня.



а) **б)** **в)**

Рис.3.3. Метод половинного ділення (а); до критеріїв досягнення збіжності (б, в)

Недоліки: повільна швидкість збіжності, знаходиться тільки один корінь на заданому інтервалі $[a,b]$. Переваги: простий, має безумовну збіжність.

3.1.2.3. Метод простих ітерацій

Представимо другу формулу (3.2) у вигляді (див. рис.3.4-а)

$$z^{(k+1)} = \vartheta(z^{(k)}) = z^{(k)} - \lambda^{(k)} \cdot f(z^{(k)}), \tag{3.4}$$

де $k = 0,1,...$ – номер ітерації. Умова збіжності ітераційного процесу:

$$0 \le \left| d\vartheta(z)/dz \right|^{(k)} \le q < 1. \tag{3.5}$$

З (3.2) легко одержати, що $d\vartheta(z)/dz = 1 - \lambda \cdot f'(z)$. Оскільки $f'(z)$ може мати позитивне, від'ємне або нульове значення, то $\lambda^{(k)}$ отримують з умови

$$|1 - \lambda^{(k)} \cdot f'(z^{(k)})| \le q < 1. \tag{3.6}$$

Якщо $q = 0$ та $f'(z^{(k)}) \ne 0$, то $\lambda^{(k)} = 1/f'(z^{(k)})$ або $\lambda^{(k)} = \lambda = 1/\max\{f'(z)\}$, $\forall z \in [a,b]$. У останньому випадку λ обирається ще до початку ітераційного процесу. Критерії досягнення збіжності – формули (3.3).

Недоліки: повільна швидкість збіжності, знаходиться тільки один корінь на заданому інтервалі $[a,b]$, потрібно обчислювати похідну. Переваги: простий.

3.1.2.4. Метод Ейткена-Стефенсена

Цей метод описується одноманітною послідовністю формул:

$$\begin{aligned}
&z^{(1)} = \varphi(z^{(0)}); \quad z^{(2)} = \varphi(z^{(1)}); \\
&z^{(3)} = z^{(0)} - \frac{\left(z^{(1)} - z^{(0)}\right)^2}{z^{(2)} - 2z^{(1)} + z^{(0)}}; \\
&z^{(4)} = \varphi(z^{(3)}); \quad z^{(5)} = \varphi(z^{(4)}); \\
&z^{(6)} = z^{(3)} - \frac{\left(z^{(4)} - z^{(3)}\right)^2}{z^{(5)} - 2z^{(4)} + z^{(3)}}; \\
&\ldots\ldots\ldots\ldots\ldots\ldots;
\end{aligned} \tag{3.7}$$

Тобто двічі застосовується метод простих ітерацій, потім проводиться корекція, що значно прискорює швидкість збіжності методу простих ітерацій. Критерії досягнення збіжності – формули (3.3), але є ще один: значне наближення до нульового значення виразу, що стоїть у знаменнику.

3.1.2.5. Метод Ньютона (метод дотичних)

З припущення, що $f(z^{(k+1)}) \approx 0$, розкладуючи цей вираз у ряд, отримують, що $f(z^{(k+1)}) \approx f(z^{(k)}) + f'(z^{(k)}) \cdot \Delta z \approx 0$, де $\Delta z = z^{(k+1)} - z^{(k)}$. Звідси (див. рис.3.4-б):

$$z^{(k+1)} = z^{(k)} - f(z^{(k)}) / f'(z^{(k)}), \quad \text{якщо } f'(z^{(k)}) \neq 0. \tag{3.8}$$

Хорошим початковим наближенням вважається таке $z^{(0)}$, при якому:

$$f(z^{(0)}) \cdot f''(z^{(0)}) > 0. \tag{3.9}$$

Л.В. Канторович довів теорему про збіжність цього методу. Згідно з нею, якщо функцію $f(z)$ можна двічі диференціювати, є такі числа $A>0$ й $B>0$, що

$$\left|1/f'(z^{(0)})\right| \le A, \quad \left|f(z^{(0)})/f'(z^{(0)})\right| \le B, \tag{3.10}$$

та виконується умова

$$|f''(z)| \le C \le 1/(2AB) \tag{3.11}$$

при будь-якому значенні z, яке задовольняє нерівності

$$|z - z^{(0)}| \le (1 - \sqrt{1 - 2ABC})/(AC), \tag{3.12}$$

то рівняння (3.1) має корінь, к якому збігається ітераційний процес (3.8), і швидкість збіжності якого оцінюється як

$$|z^{(k+1)} - z^{(k)}| \le B \cdot (2ABC)^{2^{k-1}} / 2^{k-1}. \tag{3.13}$$

Така збіжність дуже швидка (квадратична). Практично в ітераціях, як критерії досягнення збіжності, можна застосовувати формули (3.3).

Недоліки: потрібно перевіряти функцію на відповідність умовам збіжності; потрібно обчислювати значення похідної в кожній ітерації; якщо в ітерації похідна дорівнює нулю – зупинення; метод не збігається в околі кратного (не комплексного) кореня. Переваги: дуже швидкий, знаходить (якщо вони є на заданому інтервалі $[a,b]$) комплексні корені.

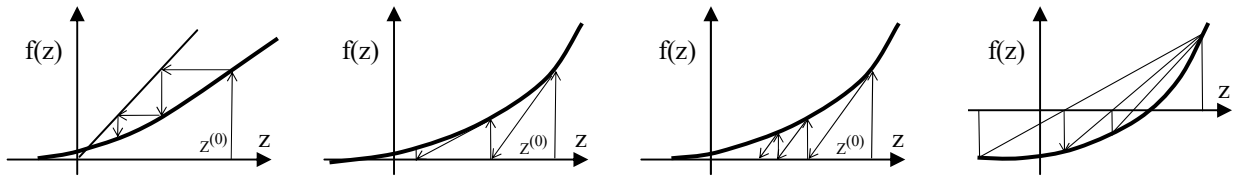


Рис.3.4. Графічне представлення методів:
а) – простих ітерацій; б) – Ньютона; в) – Ньютона модифікованого; г) – січних

Увага: якщо необхідно знайти комплексний корінь, то потрібно обирати початкове значення $z^{(0)}$ комплексним, інакше метод не буде збігатися. Крім того, дещо модифікуються умови збіжності методу: $B \le R/2$, де R – радіус кола навкруги комплексного кореня, у межах якого функція $f(z)$ – аналітична. Замість (3.11) і (3.12) – умови $2ABC = \mu \le 1$ та $|f''(z)| \le C$ при $|z - z^{(0)}| < R$. Замість (3.13) – умова $|z^{(k+1)} - z^{(k)}| \le B \cdot (\mu)^{2^{k-1}} / 2^{k-1}$.

3.1.2.6. Модифікований метод Ньютона

Якщо похідну обчислити лише в початковій ітерації, то (див. рис.3.4-в)

$$z^{(k+1)} = z^{(k)} - f(z^{(k)}) / f'(z^{(0)}), \quad \text{коли } f'(z^{(0)}) \neq 0. \tag{3.14}$$

Критерії досягнення збіжності – формули (3.3). Недоліки: потрібно обчислювати похідну; повільніший, ніж метод Ньютона, не збігається в околі кратного (не комплексного) кореня, може зовсім не збігатися. Достоїнство: простіший, ніж метод Ньютона (похідна обчислюється лише один раз).

3.1.2.7. Метод Ньютона другого порядку

Якщо прирівняти нулю суму трьох членів розкладу $f(z^{(k+1)})$ у ряд, потім за допомогою формули (3.8) виключити $\left(z^{(k+1)} - z^{(k)}\right)^2$, можна отримати таку формулу:

$$z^{(k+1)} = z^{(k)} - \frac{f(z^{(k)})}{f'(z^{(k)})} - \frac{1}{2}\frac{[f(z^{(k)})]^2 \cdot f''(z^{(k)})}{[f'(z^{(k)})]^3}, \quad \text{якщо } f'(z^{(k)}) \neq 0. \qquad (3.15)$$

Критерії досягнення збіжності – формули (3.3). Недоліки: потрібно обчислювати дві похідні, робити більшу кількість дій. Переваги: ще швидший, ніж метод Ньютона першого порядку.

3.1.2.8. Метод січних (метод хорд, спосіб пропорційних частин, правило помилкового положення)

У цьому методі кожне нове проміжне значення $z^{(k+1)}$ знаходиться як точка пересікання прямої, що з'єднує дві точки на графіку функції (3.1), причому функції в цих точках повинні бути різного знаку (див. рис.3.4-г). Математично це дається формулою:

$$z^{(k+1)} = z^{(k)} - \frac{z^{(k)} - z^{(j)}}{f(z^{(k)}) - f(z^{(j)})} \cdot f(z^{(k)}), \quad j < k, \text{ якщо } f(z^{(k)}) \cdot f(z^{(j)}) < 0. \quad (3.16)$$

Критерії досягнення збіжності – формули (3.3). Недоліки: посередня швидкість збіжності, находиться тільки один корінь на заданому інтервалі $[a,b]$. Переваги: простий.

3.1.3. Ознака (теорема) збіжності

Для всіх ітераційних методів знаходження коренів рівняння (3.1) сформульовано та доведено теорему збіжності:

Якщо існує число $M < 1$ таке, що

$$|\varphi(z_i) - \varphi(z_j)| \le M \cdot |z_i - z_j|, \qquad (3.17)$$

де будь-які дві точки z_i, $z_j \in [a,b]$; а також якщо інтервал $[a,b]$ містить значення $z^{(0)}$, $z^{(1)}$ та всі інші z, що задовольняють нерівності

$$|z - z^{(k)}| \le \frac{M}{1-M} \cdot |z^{(k)} - z^{(k-1)}| \qquad (3.18)$$

для кожного $k \ge 1$, то ітераційний процес збігається до деякого розв'язку z^* – кореня рівняння (3.1), що є єдиним в інтервалі $[a,b]$. При цьому умова (3.18) дає верхню оцінку похибки знаходження кореня.

3.2. Знаходження коренів алгебраїчних рівнянь

Якщо в рівнянні (3.1) функція $f(z)$ – алгебраїчне рівняння n-ої степені, то для відділення та знаходження коренів рівняння (3.1), окрім описаних у підрозділі 3.1, існує ще значна кількість теорем та методів, які дозволяють відділяти корені, виявляти кількість комплексних коренів, кратних коренів, знаходити корені. Далі розглянемо не всі (їх багато), а тільки деякі з них.

3.2.1. Деякі важливі теореми та формули алгебри

Алгебраїчне рівняння (АР) n-ої степені має вигляд:

$$P_n(z) = a_0 z^n + a_1 z^{n-1} + a_2 z^{n-2} + ... + a_{n-1} z + a_n = 0, \quad (3.19)$$

де в загальному випадку $z = x + iy$ – комплексні числа; a_i – дійсні числа, причому $a_0 \neq 0$.

Для швидкого обчислення полінома $P_n(z)$ послідовно виконуються такі дії:

$$P_1(z) = a_0 z + a_1; \;\; P_2(z) = P_1(z) \cdot z + a_2; \;\; ..... \;\; P_n(z) = P_{n-1}(z) \cdot z + a_n. \quad (3.20)$$

Основна теорема алгебри: алгебраїчне рівняння n-ої степені (3.19) має рівно n коренів: дійсних, комплексних та/або кратних.

Теорема 1. Комплексні корені АР (3.19) – попарно спряжені.

Наслідок. АР непарної степені має хоча б один дійсний корінь.

Теорема 2. Абсолютне значення j-го кореня лежить у межах кола радіуса R, що визначається формулою

$$|z_j^*| < 1 + \frac{A}{|a_0|} = R > 0, \quad (3.21)$$

де $A = \max\{|a_1|, |a_2|, .., |a_n|\}$.

Наслідок. Абсолютне значення j-го кореня лежить за межами кола радіуса r, що визначається формулою

$$|z_j^*| > \frac{1}{1 + \frac{B}{|a_n|}} = r \geq 0, \quad (3.22)$$

де $B = \max\{|a_0|, |a_1|, .., |a_{n-1}|\}$.

Тобто усі корені АР (3.19) лежать в межах колового кільця: $0 \leq r < |z_j^*| < R$.

Теорема Лагранжа: верхня границя *позитивних дійсних* коренів x^{*+} АР (3.19)

$$R = 1 + \sqrt[m]{\frac{B}{a_0}} > 0, \quad (3.23)$$

де $a_0 > 0$; B – максимальне значення з абсолютних величин від'ємних коефіцієнтів полінома $P_n(z)$; m – номер першого (за порядком) із від'ємних коефіцієнтів полінома $P_n(z)$.

Наслідок. Якщо поліном $P_n(z)$ не має від'ємних коефіцієнтів, то всі його корені – від'ємні.

Приклад 3.3. Маємо поліном $P(z) = -2z^4 - 3z^3 + 8z - 7 = 0$. Щоб було $a_0 > 0$, помножимо його на -1: $P(z) = 2z^4 + 3z^3 - 8z + 7 = 0$. Тоді $m = 3$, $B = |-8| = 8$, $a_0 = 2$, $x^{*+} < R = 1 + \sqrt[3]{8/2} \approx 2.5874$. Однак цей поліном не має позитивних коренів, хоча і має локальний мінімум приблизно при $z \approx 0.735$.

Висновок: наявність верхньої границі позитивних дійсних коренів відповідно до формули (3.21) не гарантує наявності дійсних коренів.

3.2.2. Методи обчислення кількості дійсних коренів алгебраїчного рівняння

Теорема Гюа (ознака дійсності всіх коренів АР): якщо в АР (3.19) усі коефіцієнти – дійсні, то для всіх $k = 1,2,...,n-1$:

$$a_k^2 > a_{k-1} \cdot a_{k+1}. \tag{3.24}$$

Теорема Декарта: кількість дійсних позитивно визначених коренів x^{*+} (з урахуванням кратності) дорівнює кількості змін знаків коефіцієнтів $a_0, a_1, ..., a_n$, або менше цієї кількості на парне число.

Крім описаних є ще декілька методів обчислення кількості дійсних коренів алгебраїчного рівняння, зокрема, основаних на теоремах **Штурма** та **Бюдана-Фур'є**. Всі ці методи потребують багато додаткових обчислень, особливо – метод Штурма.

Приклад 3.4. Є поліном $P_4(z) = z^4 + 8z^3 - 12z^2 + 104z - 20 = 0$. Потрібно визначитися із кількістю дійсних та комплексних коренів. Згідно з основною теоремою алгебри поліном має 4 кореня. При застосуванні теореми Гюа виявляється, що $(-12)^2 < 8 \cdot 104$, тому поліном має комплексні корені (2 або всі 4). Кількість змін знаків коефіцієнтів полінома дорівнює трьом, тобто, згідно з теоремою Декарта, поліном має 3 або 1 дійсних позитивно визначених коренів x^{*+}. Отже, поліном має 2 комплексних та 2 дійсних кореня, причому один з дійсних не менше нуля, а інший – від'ємний.

3.2.3. Про спеціальні методи знаходження коренів алгебраїчних рівнянь

Розроблено декілька спеціальних методів знаходження коренів алгебраїчних рівнянь. Серед них такі методи: Лобачевского-Греффе, Горнера, матричний, Мюллера, Берстоу, Бернуллі, алгоритми розділених різниць.

Доказана теорема, що для поліномів степенів, вищих за 4, не можна побудувати алгоритм зі скінченною кількістю дій, що використовує тільки прості математичні операції та обчислення коренів чисел. Інакше кажучи, для них алгоритми можуть бути тільки ітераційними.

Оскільки за допомогою програм Excel, MathCAD, Matlab, Scilab, Mathematica, Maple та інших можна переглянути графіки практично будь-яких функцій у будь-якому діапазоні, то в останній час графічний метод відділення коренів та подальшого їх знаходження фактично витіснив з інженерної практики знаходження коренів алгебраїчних рівнянь вказані спеціальні методи. З цих причин вони тут не розглядаються.

Контрольні питання до підрозділу 3.1

1. Які значення аргументів функції називають її коренями?
2. Про що стверджує теорема Коші?
3. Який алгоритм рекомендують для відділення коренів?
4. Що характерно для застосування графічного методу відділення коренів та їх знаходження?
5. Ідея, недоліки та переваги методу половинного ділення.
6. Чи може метод простих ітерацій знайти декілька коренів на заданому інтервалі?
7. Який метод має за основу метод Ейткена-Стефенсена?
8. Ідея, недоліки та переваги методів Ньютона.
9. Ідея, недоліки та переваги методу січних.
10. Яку оцінку похибки знаходження кореня трансцендентної функції дає теорема збіжності?

Контрольні питання до підрозділу 3.2

1. Як формулюється основна теорема алгебри про алгебраїчне рівняння n-ої степені?
2. Про що стверджує теорема Лагранжа та які наслідки вона має?

61 Про що стверджують теореми Гюа та Декарта?

Розділ 4

ОСНОВНІ ВЛАСТИВОСТІ ЧИСЛОВИХ МАТРИЦЬ

Числова матриця (далі просто матриця) – звичайний об'єкт чисельних методів. Зокрема, квадратна матриця присутня в будь-якій системі алгебраїчних рівнянь (САР). Саме до них зводяться практично всі чисельні методи розв'язування крайових задач. При цьому матриці САР можуть мати багато та дуже багато рядків і стовпців, а щільність їх заповнення ненульовими значеннями зазвичай відносно незначна. Це потребує розробки спеціальних методів упорядковування ("пакування") великих розріджених матриць і спеціальні алгоритми розв'язування систем рівнянь з такими матрицями.

4.1. Види числових матриць

Числовий масив $[A]$, який має вигляд двовимірної таблиці, тобто є сукупністю M рядків та N стовпців чисел, називається матрицею розмірністю $M \times N$ з елементами a_{mn}. Може мати комплексні числа (не розглядаємо).

Матрицю можна вважати лінійним оператором (див. Розділ 2). Тому багато її властивостей, які можуть відображатися на її назві, такі ж самі, як й у лінійного оператора.

Матрицю називають:

- *квадратною* (при $M = N$), прямокутною (при $M \neq N$), матрицею-рядком (при $M = 1$), матрицею-стовпцем або просто вектором $\{x\}$ (при $N = 1$);
- *розрідженою* (коли значна кількість елементів $a_{mn} = 0,\ m \neq n$), верхньою трикутною (коли $M = N$ та всі $a_{mn} = 0$ при $m > n$); нижньою трикутною (коли $M = N$ та всі $a_{mn} = 0$ при $m < n$); діагональною (при $M = N$ та $a_{mn} = 0$ при $m \neq n$); нульовою (всі $a_{mn} = 0$);
- *транспонованою* $[A]^T$, якщо її компоненти $[a_{mn}]^T = [a_{nm}]$;
- *симетричною*, якщо $M = N$ та $a_{mn} = a_{nm}$ або $[A]^T = [A]$;
- *одиничною* $[I]$, якщо $[A][I] = [A]$ або $[I][A] = [A]$. Тому матриця $[I]$ є діагональною, з компонентами $a_{mm} = 1$ на діагоналі;
- *не виродженою* або *не особливою*, якщо існує матриця $[A]^{-1}$ така, що $[A][A]^{-1} = [I]$, або якщо $\det[A] \neq 0$;
- *позитивно визначеною*, якщо $\det[A] > 0$ або, що теж саме, для будь-якого ненульового вектора $\{x\}$ скалярний добуток $(\{x\}, [A]\{x\}) = \{x\}^T[A]\{x\} > 0$. Її діагональні елементи завжди більше нуля;
- *позитивною*, якщо $\det[A] \geq 0$;
- *оберненою* $[A]^{-1}$, якщо $[A][A]^{-1} = [A]^{-1}[A] = [I]$.

Дві матриці називають:

- *ортогональними*, якщо $[A]^T[B]=[B][A]^T=[I]$ (якщо такі матриці мають комплексні числа, то їх називають ортонормальними);
- *комутативними*, якщо $[A][B]=[B][A]$; $[A]^T=-[A]$;
- *конгруентними* (матриці $[\widetilde{A}]$ і $[A]$), якщо $[\widetilde{A}]=[B]^T[A][B]$;
- *спряженими* (матриці $[A^*]$ і $[A]$), якщо $\{y\}[A]\{x\}=\{x\}[A^*]\{y\}$;
- *симетричними* або *кососиметричними*, якщо $[A]^T=[A]$ або $[A]^T=-[A]$;
- *подібними* (матриці $[A]$ і $[B]$), якщо $[B]=[W][A][W]^{-1}$, де $[W]$ – невироджена матриця.

4.2. Характеристики векторів і матриць

Детермінант квадратної матриці $[A]$ – це дійсне число

$$\det[A]=\begin{vmatrix} a_{11} & a_{12} & \dots & a_{1N} \\ a_{21} & a_{22} & \dots & a_{2N} \\ \dots & \dots & \dots & \dots \\ a_{N1} & a_{N2} & \dots & a_{NN} \end{vmatrix}. \qquad (4.1)$$

Ранг r матриці – кількість лінійно незалежних рядків матриці, причому $r_{[A]+[B]}\le r_{[A]}+r_{[B]}$. *Слід* (*шпур*) матриці $Tr([A])=\sum_m a_{mm}$, тобто дорівнює сумі діагональних членів матриці. Модуль матриці $|[A]|=[|a_{mn}|]$, причому $|\beta[A]|=|\beta|\cdot|[A]|$; $|[A]+[B]|\le|[A]|+|[B]|$; $|[A][B]|\le|[A]|\cdot|[B]|$. *Норма* матриці $[A]$ – дійсне число $\|[A]\|$, причому $\|[A]\|\ge 0$, де $\|[A]\|=0$ лише для нульової матриці; $\|\beta[A]\|=|\beta|\cdot\|[A]\|$; $\|[A]+[B]\|\le\|[A]\|+\|[B]\|$; $\|[A][B]\|\le\|[A]\|\cdot\|[B]\|$.

Якщо для будь-якого вектора $\{x\}$ виконується умова $\|[A]\{x\}\|\le\|[A]\|\cdot\|\{x\}\|$, то норма матриці $[A]$ називається *узгодженою* з нормою вектора $\{x\}$. Якщо додатково виконується умова $\|[A]\|=\max_{\|\{x\}\|=1}\|[A]\{x\}\|$, то норма матриці $[A]$ називається *підпорядкованою* нормі вектора $\{x\}$. Нижче вказані чотири норми векторів та підпорядковані їм норми матриць:

- m-норма (кубічна): $\|\{x\}\|_m=\max_i|x_i|$; $\|[A]\|_m=\max_i\sum_j|a_{ij}|$;
- l-норма (октаедрічна): $\|\{x\}\|_l=\sum_i|x_i|$; $\|[A]\|_l=\max_j\sum_i|a_{ij}|$;
- p-норма: $\|\{x\}\|_p=\sqrt[p]{\sum_i|x_i|^p}$; $\|[A]\|_p=\sqrt[p]{\sum_i\sum_j|a_{ij}|^p}$;
- k-норма (евклідова, сферична, Фробеніуса): $\|\{x\}\|_k=\sqrt{\sum_i x_i^2}$; $\|[A]\|_k=\sqrt{\sum_i\sum_j(a_{ij})^2}$. Часто замість k пишуть індекс 2.

Крім того, використовуються й інші норми, зокрема, такі норми матриць:

- s-норма: $\|[A]\|_s = \sum_i \sum_j |a_{ij}|$;
- ξ-норма: $\|[A]\|_\xi = \max |\sum_i \sum_j a_{ij}\xi_i\eta_j|$, де $\sum_i |\xi_i|^2 = \sum_j |\eta_j|^2 = 1$;
- w-норма: $\|[A]\|_w = \|[W][A][W]^{-1}\|$, де $[W]$ – будь-яка не вироджена матриця.

Власними значеннями (*характеристичними числами*) квадратної матриці $[A]$ називаються ті значення скалярного параметра λ, для яких матриця $[A]-\lambda[I]$ є виродженою. *Спектром* матриці називається вся множина її власних значень. Докладно про алгоритми знаходження власних значень та спектру матриці див. у Розділі 6.

4.3. Основні операції з матрицями

Основні операції з матрицями:

- *порівняння* двох матриць $[A]$ та $[B]$: перевіряється, чи всі $a_{mn} = b_{nm}$;
- *додавання* двох матриць $[A]$ та $[B]$: результат $[A]+[B]=[C]$, де компоненти $c_{mn} = a_{mn} + b_{mn}$;
- знаходження добутку скаляру β та матриці $[A]$: результат $\beta[A]=[C]$, де компоненти $c_{mn} = \beta a_{mn}$;
- знаходження добутку двох матриць $[A]$ та $[B]$: результат $[A][B]=[C]$, де компоненти $c_{mn} = \sum_{k=1}^{N} a_{mk}b_{kn}$;
- знаходження похідної від матриці: $d[A(t)]/dt \equiv \left[da_{mn}(t)/dt\right]$.

Як наслідок, можна отримати наступні основні співвідношення

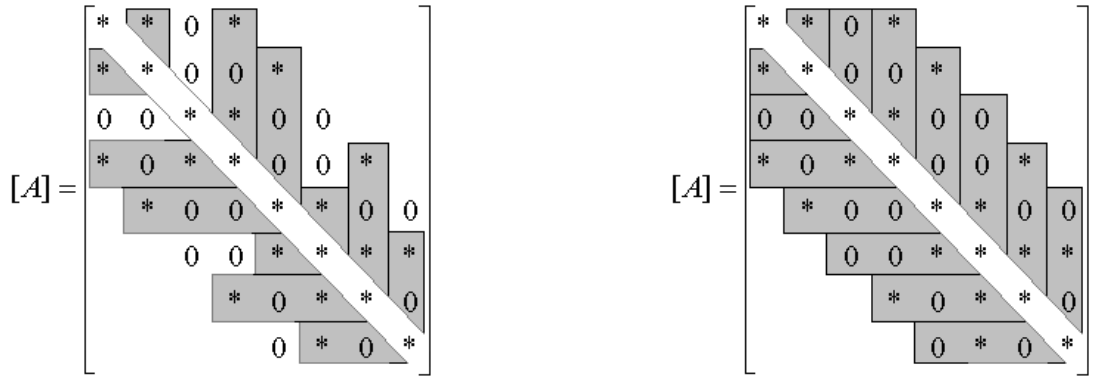
- для довільних співвідношень між M та N (зокрема, і для векторів):
  - $[A]+[B]=[B]+[A]$; $[A]+([B]+[C])=([A]+[B])+[C]$;
  - $\alpha(\beta[A])=(\alpha\beta)[A]$;
  - $\beta([A]+[B])=\beta[A]+\beta[B]$; $(\alpha+\beta)[A])=\alpha[A]+\beta[A]$;
  - $\left([A]^T\right)^T=[A]$; $\left([A]+[B]\right)^T=[A]^T+[B]^T$;
  - матриця $[B]=[A]^T[A]$ є симетричною;
- для матриць, що погоджені за кількістю рядків та стовпців:
  - $\beta([A][B])=(\beta[A])[B]=[A](\beta[B])$;
  - $[A]([B]+[C])=[A][B]+[A][C]$; $([B]+[C])[A]=[B][A]+[C][A]$;
  - $[A]([B][C])=([A][B])[C]$;
  - $[A][B]\neq[B][A]$ у загальному випадку (відсутність комутативності);
  - $\left([A][B]\right)^T=[B]^T[A]^T$;
- для квадратних матриць:
  - $\det(\beta[A])=\beta^N\det[A]$; $\det[A]^T=\det[A]$;
  - $\det[A]^{-1}=1/\det[A]$; $\left([A][B]\right)^{-1}=[B]^{-1}[A]^{-1}$; $\left([A]^{-1}\right)^T=\left([A]^T\right)^{-1}$.

4.4. Про упорядковування великих "розріджених" матриць

Коли "розріджена" матриця є матрицею системи лінійних алгебраїчних рівнянь (СЛАР), то кількість дій, необхідних для отримання розв'язку цієї СЛАР одним з прямих методів (див. Розділ 5), залежить від того, яким чином розташовані в ній ненульові компоненти. Ця залежність є нелінійною, тому кількість дій швидко збільшується при збільшенні розмірів СЛАР.

Розрідженість матриць СЛАР породжується різницевими та проєкційно-сітковими методами апроксимації крайових задач, в яких є рівняння з похідними. При використанні методу скінченних різниць (див. Розділ 15) у *кожному* вузлі застосовується шаблон вузлів (цей вузол і декілька суміжних), на основі якого проводиться наближення похідної. Кожна незаборонена ступінь свободи вузла додає в СЛАР невідому, тобто рядок (та стовпець) в матрицю СЛАР, номер якого відповідає номеру вузла $n = 1, 2, ...N^K$ (K – від англ. **Knot**: вузол) та номеру його ступені свободи $j = 1, ...J_n$. Загальна кількість невідомих у СЛАР дорівнює $N = \sum_{n=1}^{N^K} J_n$. В рядку (у стовпці) матриці будуть ненульовими лише ті компоненти матриці, які відповідають вузлам *шаблону* та його ступеням свободи. Оскільки вузли мають єдину для всього тіла нумерацію, то в шаблон (зазвичай від 2-х до 7 вузлів) можуть входити вузли, номери яких дуже відрізняються. Тоді ці ненульові компоненти рядка (стовпця) матриці СЛАР є "розкиданими" в рядку (у стовпці). Матриця – розріджена. Майже подібна ситуація – при використанні методу скінченних елементів (див. Розділ 19 та наступні). Тільки замість шаблону вузлів використовуються скінченні елементи, кожний з яких теж має обмежену кількість вузлів з аналогічними властивостями.

Профілі матриці – простори від діагоналі матриці до границь розташування ненульових компонент матриці (див. рис.4.1-а). При цьому профіль нижньої лівої частини матриці зазвичай розглядається як складений *рядками*, а верхньої правої – *стовпцями* (або навпаки). Діагональні компоненти матриці в профілі не включаються. Якщо матриця – симетрична, то обидва профілі – однакові.



а) **б)**

Рис.4.1. Точний (а) і стрічковий (б) симетричні профілі матриці (* - ненульові компоненти матриці)

Профіль матриці, довжина основних рядків (стовпців) якого є однаковою (див. рис.4.1-б), називається *стрічковим*.

Всередині профілю може бути багато або дуже багато нулів, причому ці нульові компоненти у процесі розв'язування СЛАР зазвичай змінюють свої значення на ненульові (явище *заповнення* профілю). Чим меншим є розмір профілю, тим менше компонент матриці бере участь у математичних діях, тим менше цих дій, тим швидше буде отримано результат.

Як виявилося, щоб прискорити розв'язування СЛАР прямими методами, можна особливим чином "підібрати" порядок нумерації вузлів. Як результат зміни нумерації вузлів отримують так звану матрицю перестановки $[P]$. СЛАР $[A]\{x\}=\{b\}$ після модифікації буде мати вигляд:

$$([P][A][P]^T)([P]\{x\})=([P]\{b\}); \quad [\tilde{A}]\{\tilde{x}\}=\{\tilde{b}\}, \tag{4.2}$$

де позначені $[\tilde{A}]=[P][A][P]^T$, $\{\tilde{x}\}=[P]\{x\}$, $[\tilde{b}]=[P]\{b\}$. Задача оптимізації не є тривіальною, більш того, як з'ясувалося, не має єдиного варіанта розв'язку. Можна отримати лише деяке наближення до ідеалу. Докладні відомості про методи розв'язування цієї проблеми є, зокрема, у книзі [17]. У ній застосовується теорія графів, зв'язані списки суміжності для них, стрічкові та профільні схеми упорядкування матриць і розв'язування СЛАР. Є й обмеження: розглядаються СЛАР тільки з позитивно визначеними симетричними матрицями, використовується метод квадратних коренів (див. п.5.3.5), але розширення розглянутих алгоритмів упорядкування на випадок несиметричних матриць та застосування загального методу Холецького (див. п.5.3.6) досить тривіальне.

Зазвичай мінімізують відносну кількість вихідних ненульових компонент профілю матриці:

$$\eta([L])=N+\sum_{m=1}^{N-1}\eta(v_m), \tag{4.3}$$

де оператор $\eta()$ визначає кількість ненульових елементів об'єкту; $[L]$ – нижня трикутна матриця представлення матриці СЛАР у вигляді $[A]=[L][L]^T$ (див. п.5.3.5); N – кількість невідомих у СЛАР; v_m – піддіагональна частка m-го рядка матриці $[L]$ з максимально можливою довжиною $N-m$. Це роблять тому, що доведено теорему, що кількість $f([L])$ простих математичних операцій (+, -, *, /) для отримання матриці $[L]$

$$f([L])\approx\frac{1}{2}\sum_{m=1}^{N-1}[\eta(L_{*m})-1][\eta(L_{*m})+2], \tag{4.4}$$

де L_{*m} – m-й стовпець матриці $[L]$. Для порівняння: для повністю заповненої матриці $f([L])\approx N^3/3+N^2$.

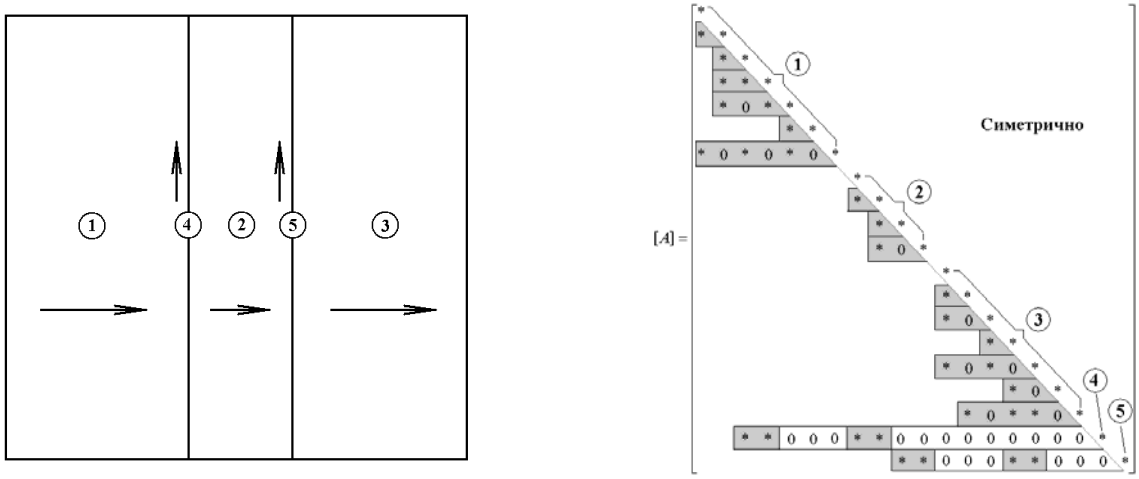
Оскільки пошук оптимального профілю може займати багато часу, то навіть стрічковий профіль матриці (див. рис.4.1-б) іноді може застосовуватися як альтернативне перше наближення до оптимального профілю.

У книзі [17] докладно розглянуті такі методи пошуку оптимального профілю: RCM – зворотний алгоритм Катхілла-Маккі; RQT – алгоритм рафінованого деревоподібного впорядковування; 1WD – алгоритм паралельних перетинів; QMD – алгоритм мінімальної степені з факторизацією; ND – алгоритм вкладених перетинів, причому алгоритм RCM входить у склад інших алгоритмів.

Візуально результат оптимізації може виглядати або як звуження *всього* профілю (концентрація ненульових компонент навколо діагоналі) або звуження *частки* профілю при одночасному підвищенню відносного заповнення "довгих" рядків (стовпців) профілю, або у перетворення матриці на блочну структуру (алгоритм 1WD) з оптимізованими профілями блоків.

Ефективність алгоритмів дуже висока і приблизно однакова, але автори книги виділяють алгоритм 1WD (паралельних перетинів).

У цьому алгоритмі все тіло розсікається на блоки, не обов'язково однакового розміру (див. рис.4.2-а), кожний з яких має свою множину вузлів (блоки *основних* вузлів), причому "граничні" вузли, які є загальними для двох (або більшої кількості) блоків, теж виділяють у окремі блоки (*вузлів-роздільників*). У кожному з блоків за допомогою алгоритму RCM (або іншого) знаходиться свій оптимальний профіль.



а) **б)**

Рис.4.2. До алгоритму 1WD (паралельних перетинів): а) – розсічення тіла на блоки, їхня нумерація (стрілки – напрямок нумерації вузлів блоків); б) – вихідні профілі блоків матриці ("білі" нулі у профілях блоків 4 й 5 – зекономлена пам'ять при блочному зберіганні матриці)

Але основний ефект досягається завдяки застосуванню техніки блочних матриць, коли матриця зберігається та обробляється блоками. Справа у тому, що "заповнення профілю" при розв'язуванні СЛАР спостерігається не на всій довжині рядка профілю, а тільки на загальних для двох блоків периферійних частинах рядка (див. рис.4.2-б). Коли блоків – небагато, то цей факт можна використовувати з відносно невеликими витратами часу. *Для тієї частини профілю, яка не заповнена споконвічно та не буде заповнюватися при розв'язуванні СЛАР, пам'ять можна зовсім не виділяти*. Ефект економії пам'яті у цьому алгоритмі різко підсилюється при збільшенні розмірів СЛАР, а також для масивних тіл з приблизно однаковими розмірами у всіх напрямках.

Якщо тіло виглядає як складна структура з *відгалуженими* частками, вся множина вузлів поділяється на групи вузлів, які відповідають кожній "гілці" тіла. Цей факт добре використовують алгоритми RQT та ND.

Ще додамо, що в алгоритмі оптимізації профілю RCM спочатку знаходиться так званий *псевдопериферійний* вузол, з якого найкраще починати нумерацію вузлів тіла або окремих його блоків.

Контрольні питання до підрозділу 4.1

1. Який об’єкт називають числовою матрицею?
2. Яким чином можна визначитися, чи є матриця позитивно визначеною?
3. Які матриці називають конгруентними?

Контрольні питання до підрозділу 4.2

1. Чому може дорівнювати детермінант квадратної матриці?
2. Які існують норми матриці?
3. Якій основній умові відповідають власні значення (характеристичні числа) квадратної матриці?

Контрольні питання до підрозділу 4.3

1. Як знайти похідну від матриці?
2. Чи є матриці лінійними операторами?

Контрольні питання до підрозділу 4.4

1. Як виникають "розріджені" матриці?
2. Для чого проводять упорядковування великих "розріджених" матриць?
3. Чи має задача оптимізації нумерації вузлів єдиний варіант розв’язку?

Розділ 5

ПРЯМІ МЕТОДИ РОЗВ'ЯЗУВАННЯ СИСТЕМ ЛІНІЙНИХ АЛГЕБРАЇЧНИХ РІВНЯНЬ

5.1. Загальні зауваження

Маємо таку задачу: є система лінійних алгебраїчних рівнянь (СЛАР), яку необхідно розв'язати:

$$[A]\{x\} = \{b\}, \quad \text{або в індекній формі} \quad a_{mn}x_n = b_m, \tag{5.1}$$

де $m,n=1,2,...,N$; $N>1$ – кількість рівнінь (невідомих) у СЛАР.

Всі методи розв'язування СЛАР (5.1) ділять на такі групи:

- *прямі*, коли розв'язок СЛАР утворюється за заздалегідь визначену кількість операцій;
- *ітераційні*, коли кількість операцій для отримання розв'язку СЛАР заздалегідь не можна точно передбачити.

Крім того, методи розв'язування СЛАР зазвичай враховують такі властивості її матриці:

- матриця СЛАР – симетрична або несиметрична, визначена позитивно або ні, має нулі на діагоналі або ні;
- матриця СЛАР – розріджена (рідко заповнена) або ні.

Примітка 5.1. Будь-яку СЛАР (5.1) з несиметричною матрицею $[A]$ можна зробити системою з симетричною матрицею, якщо помножити СЛАР зліва на транспоновану матрицю $[A]$, тобто на $[A]^T$. Система отримає вигляд

$$[\widetilde{A}]\{x\} = \{\widetilde{b}\}, \quad \text{де} \quad [\widetilde{A}] = [A]^T[A],\ \{\widetilde{b}\} = [A]^T\{b\}. \tag{5.2}$$

Недоліки: потребує значну кількість дій, змінюється структура заповненості матриці СЛАР, зростає число обумовленості матриці СЛАР (див. підрозділ 5.2). Переваги: після симетризації матриці СЛАР для її збереження може використовуватися менший об'єм пам'яті; СЛАР можна розв'язувати більш ефективними методами.

Іноді для несиметричної матриці $[A]$ вдається знайти таку розріджену матрицю $[W]$, що матриця $[\overline{A}] = [W][A][W]^{-1}$ є симетричною.

Примітка 5.2. Позитивна визначеність матриці гарантує наявність її оберненої матриці, оскільки при цьому детермінант матриці не нульовий. Нижче всюди, окрім п.5.3.8, будемо вважати, що матриці СЛАР – визначені позитивно. Нагадаємо (див. підрозділ 4.1), що матриця є визначеною позитивно, якщо для будь-якого ненульового вектора $\{x\}$ скалярний добуток

$$(\{x\}, [A]\{x\}) = \{x\}^T[A]\{x\} > 0. \tag{5.3}$$

У такій матриці діагональні елементи завжди більше нуля.

Основні відомості про матриці та їх властивості викладено у Розділі 4.

5.2. Обумовленість СЛАР

Коректність вихідної задачі можна встановити, розглядаючи властивості СЛАР (5.1). Якщо обернена матриця $[A]^{-1}$ існує, то $\det[A] \neq 0$, та навпаки. Але наскільки $\det[A]$ повинен відрізнятися від нуля, щоб розв'язок СЛАР був достатньо точним? Застосовують такі оцінки:

а/ система (5.1) стійка відносно змін у *правій частині*, якщо при будь-яких варіаціях вектора $\{b\}$ (правої частини СЛАР) норма варіації розв'язку є обмеженою:

$$\|\delta\{x\}\| \le \nu_{\{b\}} \cdot \|\delta\{b\}\|, \tag{5.4}$$

де *число обумовленості* СЛАР відносно змін у *правій частині* $\nu_{\{b\}} > 0$ та не залежить від $\{b\}$;

б/ система (5.1) стійка відносно змін у *матриці*, якщо

$$\|\delta\{x\}\| / \|\{x\}\| \le \nu_{[A]} \cdot \|\delta\{b\}\| / \|\{b\}\|, \tag{5.5}$$

де *число обумовленост*і матриці

$$\nu_{[A]} = \|[A]^{-1}\| \|[A]\|; \quad \nu_{[A]} \ge 1, \tag{5.6}$$

причому застосовується будь-яка з існуючих підпорядкованих вектору норм (див. підрозділ 4.2). Число обумовленості $\nu_{[A]}$ *симетричної* матриці $[A]$, крім формули (5.6), можна обчислити як відношення максимального власного значення (характеристичного числа) матриці $[A]$ до мінімального. При дуже великих значеннях $\nu_{[A]}$ матриця $[A]$ вважається *погано обумовленою*.

в/ загальна оцінка. Залучаються всі можливі для СЛАР фактори: $\|\delta\{x\}\| / \|\{x\}\|$, $\nu_{[A]}$, $\|\delta[A]\| / \|[A]\|$ та $\|\delta\{b\}\| / \|\{b\}\|$:

$$\frac{\|\delta\{x\}\|}{\|\{x\}\|} \le \frac{\nu_{[A]}}{1 - \nu_{[A]} \cdot \|\delta[A]\| / \|[A]\|} \cdot \left(\frac{\|\delta[A]\|}{\|[A]\|} + \frac{\|\delta\{b\}\|}{\|\{b\}\|} \right). \tag{5.7}$$

Оскільки $\|\delta[A]\| / \|[A]\| = O(N \cdot 2^{-p})$, де p – кількість бітів ЕОМ для мантиси дійсного числа з формою числа з розділовим знаком, що плаває (див. таблицю 1.5), то загальна оцінка похибки розв'язку СЛАР

$$\|\delta\{x\}\| / \|\{x\}\| = O(\nu_{[A]} \cdot N \cdot 2^{-p}). \tag{5.8}$$

При $p = 53$ (формат **double**) $2^{-53} \approx 2.2204460492503131 \cdot 10^{-16}$ (машинний іпсилон), тому для одержання точності розв'язку СЛАР приблизно в 1% добуток $\nu_{[A]} \cdot N$ не повинен перевищувати числа 10^{13}.

5.3. Прямі методи розв'язування систем лінійних алгебраїчних рівнянь

Як це вже було зазначено, прямі методи розв'язування СЛАР мають заздалегідь визначену кількість операцій. Розроблено значну кількість прямих методів. Розглянемо тільки основні, які або популярні, або відомі ще з елементарної алгебри. При цьому абстрагуємося від ступені заповнення матриці СЛАР.

5.3.1. Метод використання оберненої матриці СЛАР

У цьому методі йдеться про використання оберненої матриці $[A]^{-1}$, отриманої з матриці $[A]$. Помножимо СЛАР (5.1) зліва на $[A]^{-1}$:

$$[A]^{-1}[A]\{x\}=[A]^{-1}\{b\}\ ;\quad [A]^{-1}[A]=[I]\ ;\quad [I]\{x\}=\{x\}\ ;\quad \{x\}=[A]^{-1}\{b\}. \qquad (5.9)$$

Тут $[I]$ – одинична матриця. Отже, якщо знайдена обернена матриця $[A]^{-1}$, то розв'язок СЛАР отримується застосуванням останнього виразу з (5.9). Такий метод розв'язування СЛАР майже не застосовується, оскільки є раціональніші методи. Однак іноді потрібно мати саме матрицю $[A]^{-1}$. Розглянемо лише один метод знаходження матриці $[A]^{-1}$, який зветься методом облямівки.

Нехай є матриця $[A]$ з компонентами $a_{mn}; m,n=1,...,N$, потрібно знайти обернену матрицю $[A]^{-1}$ з компонентами β_{mn}. Розглянемо послідовність матриць

$$[S]_1=\begin{bmatrix}a_{11}\end{bmatrix};\ [S]_2=\begin{bmatrix}a_{11} & a_{12}\\ a_{21} & a_{22}\end{bmatrix}=\begin{bmatrix}[S]_1 & a_{12}\\ a_{21} & a_{22}\end{bmatrix};\ [S]_3=\begin{bmatrix}a_{11} & a_{12} & a_{13}\\ a_{21} & a_{22} & a_{23}\\ a_{31} & a_{32} & a_{33}\end{bmatrix}=\begin{bmatrix}[S]_2 & a_{13}\\ a_{31} & a_{32} & a_{33}\end{bmatrix};\ \ldots;$$

$$[S]_k=\begin{bmatrix}a_{11} & \ldots & a_{1k}\\ \ldots & .. & \ldots\\ a_{k1} & \ldots & a_{kk}\end{bmatrix}=\begin{bmatrix}[S]_{k-1} & a_{1k}\\ a_{k1} & \ldots & a_{kk}\end{bmatrix};\quad \ldots;$$

$$[A]=[S]_N=\begin{bmatrix}a_{11} & \ldots & a_{1N}\\ \ldots & .. & \ldots\\ a_{N1} & \ldots & a_{NN}\end{bmatrix}=\begin{bmatrix}[S]_{N-1} & a_{1N}\\ a_{N1} & \ldots & a_{NN}\end{bmatrix}. \qquad (5.10)$$

Кожну таку $k-$ту матрицю, починаючи з $[S]_3$, можна уявити у вигляді блочної матриці з чотирма блоками:

$$[S]_k=\begin{bmatrix}[S]_{k-1} & [a_{1k}]\\ [a_{k1}] & [a_{kk}]\end{bmatrix}, \qquad (5.11)$$

причому блок $[a_{kk}]=a_{kk}$, тобо є окремим числом, блок $[a_{k1}]$ є рядком, а блок $[a_{1k}]$ – стовпцом. Обернену матрицю $[A]^{-1}$ теж представимо у блочному вигляді

$$[A]^{-1}=\begin{bmatrix}[B]_{N-1} & [\beta_{1N}]\\ [\beta_{N1}] & [\beta_{NN}]\end{bmatrix}. \qquad (5.12)$$

Якщо знайдено блок $[B]_{N-1}$, то, як буде показано, можна швидко знайти інші блокі матриці, тобто знайти матрицю $[A]^{-1}$. Тобто можна створювати послідовність обернених матриць $[B]_k;\ k=2,...,N$ аналогічно з послідовністю (5.10), остання з яких буде якраз $[A]^{-1}$.

З використанням блоків запишемо $[A]^{-1}[A]=[I]$ як

$$[A]^{-1}[A]=\begin{bmatrix}[B]_{N-1} & [\beta_{1N}]\\ [\beta_{N1}] & [\beta_{NN}]\end{bmatrix}\begin{bmatrix}[S]_{N-1} & [a_{1N}]\\ [a_{N1}] & [a_{NN}]\end{bmatrix}=[I], \qquad (5.13)$$

де $[I]$ – одинична матриця; або після перемноження блоків:

$$[B]_{N-1}[S]_{N-1}+[\beta_{1N}][a_{N1}]=[I];\quad [B]_{N-1}[a_{1N}]+[\beta_{1N}][a_{NN}]=[0]; \tag{5.14}$$

$$[\beta_{N1}][S]_{N-1}+[\beta_{NN}][a_{N1}]=[0];\quad [\beta_{N1}][a_{1N}]+[\beta_{NN}][a_{NN}]=[I]. \tag{5.15}$$

Ця система розв'язується відносно невідомих складових оберненої матриці. З (5.14), якщо спочатку виключити $[B]_{N-1}$, то можна одержати такі вирази:

$$[\beta_{1N}]=-[S]_{N-1}^{-1}[a_{1N}]\left([a_{NN}]-[a_{N1}][S]_{N-1}^{-1}[a_{1N}]\right)^{-1};\quad [B]_{N-1}=([I]-[\beta_{1N}][a_{N1}])[S]_{N-1}^{-1}; \tag{5.16}$$

Позначимо:

$$[X]=[S]_{N-1}^{-1}[a_{1N}];\quad [Y]=[a_{N1}][S]_{N-1}^{-1};\quad [\theta]=[a_{NN}]-[a_{N1}][X]=[a_{NN}]-[Y][a_{1N}]. \tag{5.17}$$

З урахуванням позначок вирази (5.16) приймуть вигляд:

$$[\beta_{1N}]=-[X][\theta]^{-1};\quad [B]_{N-1}=[S]_{N-1}^{-1}+[X][\theta]^{-1}[Y]. \tag{5.18}$$

Аналогічні дії дозволяють з (5.15) отримати вирази для останніх невідомих блоків оберненої матриці:

$$[\beta_{N1}]=-[\theta]^{-1}[Y];\quad [\beta_{NN}]=[\theta]^{-1}. \tag{5.19}$$

Оскільки матриця $[\theta]$ містить лише одну компоненту, то $[\theta]^{-1}$ обчислюється елементарно. А матриця $[S]_{N-1}^{-1}$ є фактично матрицею $[B]_{N-1}$, отриманою для попереднього члена послідовності матриць, тобто на момент використання формул (5.17) … (5.19) є відомою.

Отже, якщо визначена обернена матриця $[S]_{k-1}^{-1}$, то можна легко визначити обернену матрицю $[S]_k^{-1}$, $k=2,...,N$. Для цього потрібно застосовувати формули (5.17) … (5.19), замінив у них індекс N на індекс k.

Стартувати можна з $k=3$, оскільки обчислення оберненої матриці $[S]_2^{-1}$ можна провести за такою простою формулою:

$$[S]_2^{-1}=\frac{1}{\Delta}\begin{bmatrix} a_{22} & -a_{12} \\ -a_{21} & a_{11} \end{bmatrix}, \tag{5.20}$$

де $\Delta=a_{11}a_{22}-a_{12}a_{21}$.

Матриця $[A]$ може бути несиметричною, але не повинна бути виродженою, тоді всі проміжні матриці $[S]_k$ не є особливими.

Оскільки при обчисленні компонент матриці $[A]^{-1}$ накоплюються похибки округлення (див. п.1.1.3 та підрозділ 1.3), то вона є наближеною. Позначимо наближену обернену матрицю $[A]^{-1}$ як $[W]$ та введемо матрицю похибки

$$[\Delta]=[I]-[A][W]. \tag{5.21}$$

Доказано, що в ітераційному процесі

$$[W]_{(k+1)}=[W]_{(k)}+[W]_{(k)}[\Delta]_{(k)};\quad k=0,1,... \tag{5.22}$$

матриця $[W]_{(k+1)}$ збігається до точної оберненої матриці $[A]^{-1}$ (у (5.22) $[\Delta]_{(k)}=$ $=[I]-[A][W]_{(k)}$). За критерій припинення ітерацій рекомендують використовувати умову

$$\left\|[W]_{(k+1)}-[W]_{(k)}\right\|\le\varepsilon, \tag{5.23}$$

де ε – задана точність.

5.3.2. Формули Крамера

Розв'язок СЛАР (5.1) із застосовуванням формул Крамера утворюється як

$$x_m = \Delta_m / \Delta, \tag{5.24}$$

де Δ – детермінант матриці $[A]$; а Δ_m – детермінант видозміненої матриці $[A]$, у яку замість m-го стовпця вставлено вектор правої частини $\{b\}$. Недолік: кількість простих математичних операцій пропорційна $N!$ (факторіал від N). Тому кількість таких операцій невиправдано велика вже при $N > 3$.

5.3.3. Метод Гаусса

Метод Гаусса має ще такі назви: *метод послідовного виключення невідомих*, *схема єдиного ділення*. Розрізняють два етапи: *прямий хід* та *зворотний хід* (це властиво для багатьох методів, тому в подальшому викладенні методів на це не будемо привертати додаткової уваги).

Метод опишемо у вигляді алгоритму. **Прямий хід:**

початок циклу з $m = 1, ..., N-1$:
якщо $a_{mm} = 0$ – зупинення;
$b_m = b_m / a_{mm}$;
початок циклу з $n = m, ..., N$:
$a_{mn} = a_{mn} / a_{mm}$;
кінець циклу з n;
$a_{mm} = 1$;
початок циклу з $k = m+1, ..., N$:
початок циклу з $i = m+1, ..., N$: (5.25)
$a_{ki} = a_{ki} - a_{km} a_{mi}$;
кінець циклу з i;
$b_k = b_k - a_{km} b_m$;
кінець циклу з k;
кінець циклу з m.
$b_N = b_N / a_{NN}$; $a_{NN} = 1$.

В процесі прямого ходу утворюється трикутна матриця: всі елементи нижче діагоналі стають нульовими, на діагоналі з'являються одиниці, модифікується вектор правої частини СЛАР. При повністю заповненій матриці прямий хід потребує кількість операцій приблизно $2N^3/3$. Якщо в циклі з n ще до проведення операції ділення виконувати перемноження діагональних елементів, то після закінчення циклу утворюється величина $a_{11} \cdot a_{22} \cdot ... \cdot a_{NN} = \det[A]$.

Зворотний хід:

$x_N = b_N$.
цикл з $m = N-1, ..., 1$:

$$x_m = b_m - \sum_{n=m+1}^{N} a_{mn} x_n \ ; \tag{5.26}$$

кінець циклу з m.

Тобто зворотний хід – визначення невідомих. Він швидкісний: при повністю заповненій матриці потребує кількість операцій, пропорційну N^2.

Переваги методу: простий у реалізації, досить швидкий, матриця може бути несиметричною, модифіковані компоненти матриці можна зберігати на місці відповідних компонент вихідної матриці, зберігається "профіль" матриці.

Недоліки методу:

а/ на прямому ході модифікується вектор правої частини СЛАР, при цьому використовуються поточні значення a_{mm}, a_{kk}, b_k. Тобто метод Гаусса потребує повторення прямого ходу навіть тоді, коли при іншому векторі правої частини СЛАР її матриця лишилася незмінною;

б/ якщо на прямому ході в циклі з m виявляється, що компонента $a_{mm} = 0$ (нуль на діагоналі), то алгоритм повинен зупинитися; якщо $a_{mm} \approx 0$, то розв'язок СЛАР буде мати значну похибку;

в/ значна кількість компонент матриці всередині "профілю", які мали нульові значення, стають ненульовими, тобто ступень заповнення матриці ненульовими компонентами збільшується.

5.3.4. Метод Гаусса з вибором головного елемента

Метод ще має майже тотожну назву *методу головних елементів*.

В цьому методі в циклі з m перед циклом з n реалізується додаткова процедура знаходження елемента, який має найбільше значення (за модулем):

- з усіх рядків з номерами від m до N включно (метод Гаусса з вибором головного елемента з усієї матриці);
- з m–го рядка (метод Гаусса з вибором головного елемента з рядка);
- з m–го стовпця (метод Гаусса з вибором головного елемента зі стовпця).

Потім рядки та стовпці переставляються місцями так, щоб обраний елемент зайняв позицію a_{mm}. Ці перестановки запам'ятовуються (створюється матриця перестановок $[P]$), щоб потім повернути знайдені x_m на свої місця.

Порівняно зі "звичайним" методом Гаусса метод має такі

- *переваги*: більша точність отриманого розв'язку; метод завжди "працює", коли $\det[A] \neq 0$;
- *недоліки*: збільшена кількість операцій, ускладнена логіка алгоритму; при симетричній перестановці змінюється структура заповнення матриці, а при несиметричній симетрична матриця стає несиметричною.

5.3.5. Метод квадратних коренів

Симетричну матрицю $[A]$ (див. Примітку 5.7 наприкінці п.5.3.6) можна представити у вигляді добутку

$$[A] = [L][L]^T, \tag{5.27}$$

де матриця $[L]$ може бути несиметричною, зокрема трикутною:

$$[L]=\begin{bmatrix} L_{11} & 0 & ... & 0 \\ L_{21} & L_{22} & ... & 0 \\ ... & ... & ... & ... \\ L_{N1} & L_{N2} & ... & L_{NN} \end{bmatrix}. \tag{5.28}$$

У методі квадратних коренів на прямому ході матриця $[A]$ СЛАР (5.1) представляється у вигляді добутку двох трикутних матриць $[L]$ та $[L]^T$. Для визначення ненульових компонент L_{mn} матриці $[L]$, як результат перемноження матриць $[L]$ та $[L]^T$, маємо для кожного $m \le n \le N$ такі рівняння:

$$L_{1m}L_{1n} + L_{2m}L_{2n} + ... + L_{mm}L_{mn} = a_{mn}. \tag{5.29}$$

Ця система рівнянь має розв'язок (це й є прямий хід):

$$\begin{cases} L_{11} = \sqrt{a_{11}}; \\ L_{1n} = a_{1n} / L_{11}; \quad (1 < n \le N); \\ L_{mm} = \sqrt{a_{mm} - \sum_{k=1}^{m-1} (L_{km})^2}; \quad (1 < m \le N); \\ L_{mn} = \left(a_{mn} - \sum_{k=1}^{m-1} L_{km}L_{kn} \right) / L_{mm}; \quad (m < n \le N); \\ L_{mn} = 0; \quad (n < m). \end{cases} \tag{5.30}$$

При повністю заповненій матриці кількість операцій приблизно $N^3/3$.

Примітка 5.3. Відомо, що $\det[A] = \det[L]^T \det[L] = (\det[L])^2 = (L_{11} \cdot L_{22} \cdot .... \cdot L_{NN})^2$.

Примітка 5.4. Формули (5.30) прямого ходу мають назву алгоритму *скалярних добутків*. Є принаймні ще два варіанти таких формул: алгоритм *облямівки* та алгоритм *зовнішніх добутків*. Вони розрізняються послідовністю дій та зонами матриці, що використовуються для обчислень: підматриці, рядки, стовпці. Якщо матриця $[A]$ є не повністю заповненою (розрідженою), то застосування того чи іншого алгоритму пов'язують зі схемою зберігання матриці та методом оптимізації її профілю (див. підрозділ 4.4).

Зворотний хід методу має два кроки:

$$\{y\} = ([L])^{-1}\{b\} \;; \quad \{x\} = ([L]^T)^{-1}\{y\}. \tag{5.31}$$

В індексній формі запису формули (5.31) мають вигляд:

$$\begin{cases} y_1 = b_1 / L_{11}; \\ y_m = \left(b_m - \sum_{k=1}^{m-1} L_{km} y_k \right) / L_{mm}; \quad (m = 2,3,...,N); \\ x_N = y_N / L_{NN}; \\ x_m = \left(y_m - \sum_{k=m+1}^{N} L_{mk} x_k \right) / L_{mm}; \quad (m = N-1, N-2, ..., 1). \end{cases} \tag{5.32}$$

Оскільки матриці $[L]$ та $[L]^T$ – трикутні, то кількість операцій зворотного ходу пропорційна N^2, тобто незначна, порівняно з прямим ходом.

Переваги: при отриманні матриці $[L]$ (прямий хід) не модифікується вектор правої частини; матриця $[L]$ може зберігатися на місці вихідної матриці $[A]$; друга матриця, тобто $[L]^T$, фактично не обчислюється та не зберігається;

не зазнає змін "профіль" матриці СЛАР; кількість дій удвічі менша, ніж у методі Гаусса (це пов'язано зі симетричністю матриці СЛАР).

Недоліки: такі ж, як і описані в пунктах б/ та в/ для методу Гаусса.

5.3.6. Схема Холецького (Гаусса-Холецького)

Інша назва: компактна схема виключення. В ній на прямому ході матриця $[A]$ СЛАР (5.1) представляється у вигляді добутку двох матриць:

$$[A]=[L][U], \tag{5.33}$$

де нижня (L – від Low) і верхня (U – від Upper) трикутні матриці

$$[L]=\begin{bmatrix} L_{11} & 0 & ... & 0 \\ L_{21} & L_{22} & ... & 0 \\ ... & ... & ... & ... \\ L_{N1} & L_{N2} & ... & L_{NN} \end{bmatrix};\quad [U]=\begin{bmatrix} 1 & U_{12} & ... & U_{1N} \\ 0 & 1 & ... & U_{2N} \\ ... & ... & ... & ... \\ 0 & 0 & ... & 1 \end{bmatrix}. \tag{5.34}$$

Необхідна і достатня умова існування та єдиного варіанта розкладу (5.33): матриця $[A]$ повинна мати всі *ненульові* кутові мінори $\Delta_1=a_{11}$, $\Delta_2=\det\begin{bmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{bmatrix}$, ..., $\Delta_N=\det[A]$.

Як й у методі квадратних коренів для визначення значень ненульових компонент матриць $[L]$ та $[U]$, як результат перемноження цих матриць, маємо систему рівнянь, розв'язок яких (це й є прямий хід):

$$\begin{cases} L_{m1}=a_{m1}; & (1\le m\le N); \\ U_{1n}=a_{1n}/L_{11}; & (1<n\le m); \\ L_{mn}=a_{mn}-\sum\limits_{k=1}^{n-1}L_{mk}U_{kn}; & (1<n\le m); \\ U_{mn}=\left(a_{mn}-\sum\limits_{k=1}^{m-1}L_{mk}U_{kn}\right)/L_{mm}; & (1<m\le n). \end{cases} \tag{5.35}$$

При повністю заповненій матриці кількість операцій приблизно $2N^3/3$.

Зворотний хід має два кроки:

$$\{y\}=([L]^T)^{-1}\{b\}\ ;\quad \{x\}=([U])^{-1}\{y\}. \tag{5.36}$$

В індексній формі запису формули (5.36) мають вигляд:

$$\begin{cases} y_1=b_1/L_{11}; \\ y_m=\left(b_m-\sum\limits_{k=1}^{m-1}L_{km}y_k\right)/L_{mm}; & (m=2,3,...,N); \\ x_N=y_N; \\ x_m=y_m-\sum\limits_{k=m+1}^{N}U_{mk}x_k; & (m=N-1,N-2,...,1). \end{cases} \tag{5.37}$$

Оскільки матриці $[L]$ та $[U]$ – трикутні, то кількість операцій зворотного ходу пропорційна N^2, тобто незначна, порівняно з прямим ходом.

Переваги: при отриманні матриць $[L]$ та $[U]$ (прямий хід) не модифікується вектор правої частини; обидві матриці можуть зберігатися на місце вихідної

матриці $[A]$ (окрім одиниць на діагоналі матриці $[U]$, що й не потрібно); не зазнає змін "профіль" матриці СЛАР.

Недоліки: такі ж, як і описані в пунктах б/ та в/ для методу Гаусса.

Примітка 5.5. Відносно формул (5.35) прямого ходу див. Примітку 5.4.

Примітка 5.6. Якщо на прямому ході методу Гаусса (див. формули (5.25)) дії з компонентами правої частини СЛАУ не проводити, а потрібні для цих дій коефіцієнти зберігати як компоненти нової матриці, то фактично отримаємо матрицю $[L]$ схеми Холецького. Крім того, матриця $[A]$ вихідної СЛАР після прямого ходу методу Гаусса точно відповідає матриці $[U]$ схеми Холецького. Отже, схема Холецького фактично є реалізацією методу Гаусса, в якій дії з компонентами правої частини СЛАУ перенесені до зворотного ходу.

Примітка 5.7. Метод квадратних коренів (див. п.5.3.5) є окремим випадком схеми Холецького, коли матриця СЛАР (5.1) – симетрична. Тому необхідна і достатня умова існування та єдиного варіанта розкладу (5.27) така ж, як і для розкладу (5.33).

5.3.7. Схема Холецького з діагональною матрицею

Якщо матриця $[A]$ – симетрична, то з формул (5.34) випливає, що $U_{mn} = L_{nm} / L_{mm}$ (за індексом m не додавати). Звідсіля $L_{mn} = U_{mn} L_{mm}$, тобто можна не збирати матрицю $[L]$ або $[U]$. Якщо збирати тільки матрицю $[L]$, то кількість операцій на прямому ході зменшиться на величину, пропорційну N^2, тобто відносно мало. Але на зворотному – збільшится на ту ж величину, що складе біля 30 відсотків дій цього ходу. Тому можна обчислювати лише компоненти матриці $[U]$ та *діагональні* компоненти матриці $[L]$, причому останні можна зберігати у матриці $[U]$ замість одиниць на діагоналі. Замість формули (5.33) записують іншу:

$$[A] = [U]^T [W][U], \tag{5.38}$$

де матриця $[W]$ – диагональна, з компонентами $W_{mm} = L_{mm}$ (див. формули (5.35)). Компоненти L_{mn} підраховуються, використовуються, але не запам'ятовуються.

Зворотний хід методу має три кроки:

$$\{y\} = ([U]^T)^{-1}\{b\}; \quad \{z\} = ([W])^{-1}\{y\}; \quad \{x\} = ([U])^{-1}\{z\}. \tag{5.39}$$

В індексній формі запису ці формули мають вигляд:

$$\begin{cases} y_1 = b_1 / W_{11}; \\ y_m = \left(b_m - \sum\limits_{k=1}^{m-1} U_{km} y_k \right) / W_{mm}; \quad (m = 2,3,...,N); \\ x_N = y_N; \\ x_m = y_m - \sum\limits_{k=m+1}^{N} U_{mk} x_k; \quad (m = N-1, N-2,...,1). \end{cases} \tag{5.40}$$

5.3.8. Ортогоналізація Грама-Шмідта

Обов'язковість $\det[A] \neq 0$ в СЛАР (5.1) – природна вимога, оскільки інакше система має багато розв'язків. Однак системи з $\det[A] = 0$ іноді збираються,

причому є бажання в автоматичному режимі виявити цей факт, знайти та виключити лінійно-залежні рядки СЛАР (це й є причиною того, що $\det[A]=0$) та зайві невідомі із системи, отримати її розв'язок для тих невідомих, що залишилися. Наприклад, ця ситуація часто виникає, коли за допомогою методу найменших квадратів проводиться аналітична апроксимація функції, заданої таблицею, яка формується в автоматичному режимі при умовах змінної бази для її формування. Є й інші ситуації.

Процес ортогоналізації Грама-Шмідта заснований на таких ідеях, фактах:

- матриця СЛАР (5.1) вироджена, коли в СЛАР є хоча б один рядок, лінійно залежний від якогось іншого;
- рядок компонент матриці СЛАР (розміром $N\times N$), після множення на вектор змінних (неважливо, що до розв'язання СЛАР він ще не відомий), може розглядатися як вектор, який має базис із N векторів, а відповідна рядку компонента правої частини – як довжина (норма) цього вектора;
- два вектори, що отримані з *лінійно залежних* рядків СЛАР, паралельні, тобто відрізняються лише довжиною (масштабом, абсолютною величиною, нормою). Якщо, наприклад, взяти два вектори та сумістити їхні початки, один з векторів назвати "головним", розкласти другий вектор на напрямок "головного" та перпендикулярно до нього, то відсутність перпендикулярної складової вказує на лінійну залежність векторів;
- якщо відкинути лінійно-залежну (паралельну) частину вектора, залишиться тільки та його частина, що ортогональна "головному" вектору. "Вилучення" повинне відобразитися й на довжині вектора, тобто на величині відповідної вектору (рядку) компоненти правої частини СЛАР. При цьому базис не модифікується, тобто розв'язок СЛАР, з рядка якої "вилучена" лінійно залежна (стосовно "головного" рядка) частина, не зміниться.

Крім виявлення рядків СЛАР, що призводять до її виродження, є потреба в розв'язанні частини СЛАР, що залишилася. Для вирішення цієї проблеми необхідно мати на увазі, що:

- якщо в СЛАР всі рядки будуть ортогональні один одному (а це можливо, тому що кількість рядків і розмір базису збігаються) і нормовані, то матриця отриманої СЛАР $[\widetilde{A}]\{x\}=\{\widetilde{b}\}$ є ортогональною, тобто має таку властивість: $[\widetilde{A}]^T[\widetilde{A}]=[I]$, де $[I]$ – одинична матриця. Інакше кажучи, $[\widetilde{A}]^T=[\widetilde{A}]^{-1}$, тобто матриця СЛАР, яка пройшла процедури ортогоналізації, нормування й транспонування, є оберненою, і знайдений вектор $\{x\}=[\widetilde{A}]^T\{\widetilde{b}\}$.
- процедури ортогоналізації та нормування СЛАР можна поєднати, а процедура транспонування – тривіальна (не вимагає обчислень);
- розв'язок можна одержати тільки для лінійно незалежної частини СЛАР, відкинувши "браковану" частину (рядки та відповідні їм стовпці).

Алгоритм може реалізовуватися таким чином:

а/ спочатку всі рядки – "непридатні";

б/ з-посеред "непридатних" рядків, що залишилися, шукаємо "головний". Починаємо цикл з "непридатних" рядків ($m = 1,2,...,N$; "придатні" рядки пропускаємо):

- обчислюємо $q = \sum_{n=1}^{N} (a_{mn})^2$;
- якщо $q > \varepsilon$ (ε – припустима точність на лінійну залежність рядків), то рядок – "придатний". Нормуємо його: $a_{mn} = a_{mn} / \sqrt{q}$, де $n = 1,2,...,N$; $b_m = b_m / \sqrt{q}$. Перериваємо цикл з "непридатних" рядків, оскільки знайдений "придатний" рядок з номером m стає "головним".

в/ з "непридатних" рядків, що залишилися, будуємо рядки (вектори), ортогональні "головному" m-му:

- починаємо цикл з "непридатних" рядків ($k = m+1,\ m+2,\ ...,\ N$; "придатні" рядки пропускаємо);
- обчислюємо $c = \sum_{n=1}^{N} a_{kn} \cdot a_{mn}$, потім $a_{kn} = a_{kn} - c \cdot a_{mn}$, де $n = 1,2,...,N$, а також $b_k = b_k - c \cdot b_m$.

г/ повертаємося у пункт б/. Якщо в ньому цикл вичерпався, але рядка з $q > \varepsilon$ (тобто "придатного") не знайшлося, то процес ортогоналізації й одночасного нормування закінчено. Компоненти всіх "непридатних" рядків стали практично нульовими.

д/ можна провести перевірку ортогональності (беруть участь тільки "придатні" рядки та відповідні їм стовпці): $a_{mk} \cdot a_{kn} = \delta_{mn}$.

е/ проводимо розв'язування СЛАР (беруть участь тільки "придатні" рядки та відповідні їм стовпці): $x_m = (a_{mn})^T \cdot b_n = a_{nm} \cdot b_n$.

Недоліки процедури: її необхідно проводити декілька разів (щоб вилучити відповідні "непридатним" рядкам стовпці), доки не стабілізуються "придатні" рядки; змінюється структура заповнення матриці СЛАР; кількість дій значно більше, ніж у методі Гаусса.

5.3.9. Порівняльна характеристика прямих методів розв'язування СЛАР

Характерні властивості прямих методів розв'язування СЛАР наведені в таблиці 5.1. В ній позначені методи та схеми: **I** – Гаусса; **II** – Гаусса з вибором головного елемента; **III** – квадратних коренів; **IV** – Холецького; **V** – Холецького з діагональною матрицею; **VI** – ортогоналізації Грама-Шмідта.

Примітка 5.8. Для всіх методів і схем характерно, що нульові компоненти матриці $[A]$ всередині "профілю" можуть стати ненульовими, тобто щільність заповнення матриці ненульовими компонентами підвищується.

Таблиця 5.1. Характерні властивості прямих методів і схем розв'язування СЛАР

Властивості \ Методи, схеми	**I**	**II**	**III**	**IV**	**V**	**VI**
обов'язковість $\det[A] \neq 0$	+	+	+	+	+	-
обов'язкова симетричність матриці $[A]$	-	-	+	-	+	-
швидкість (+ – добра; ++ – дуже добра; +++ – найшвидший)	++	++	++	++	+++	+
підвищена точність	-	+	-	-	-	-
простота у реалізації	+	-	+	+	+	-
не потребує додаткової пам'яті для збереження модифікованих матриць	+	+	+	+	+	+
вектор $\{b\}$ на прямому ході не модифікується	-	-	+	+	+	-
зупинка алгоритму при виявленні $a_{mm} = 0$	+	-	+	+	+	-
зберігається "профіль" матриці	+	-	+	+	+	-

5.4. Схеми розв'язування систем лінійних алгебраїчних рівнянь особливого вигляду: з тридіагональною матрицею

Є деякі математичні проблеми, які призводять до створення СЛАР з особливим виглядом матриці. Розглянемо лише один з варіантів, коли матриця має вигляд тридіагональної. Це характерно для розв'язування одновимірних крайових задач з похідною другого порядку методом скінченних різниць. Застосовується схема виключення Гаусса, яка традиційно записується у вигляді схеми *прогонки*.

Отже, в симетричній матриці $[A]$ нульовими компонентами a_{mn} є всі, що мають індекси $n < m-1$ та $n > m+1$. Якщо ввести фіктивні $a_{10} = a_{N,N+1} = 0$ та x_0 і x_{N+1}, а також позначити

$$\beta_m = a_{m,m-1}; \quad \gamma_m = a_{mm}; \quad \delta_m = a_{m,m+1}; \quad m = 1,2,...,N, \tag{5.41}$$

то рядки СЛАР можна записати у вигляді (фіктивні x_0 та x_{N+1} подавляються нульовими коефіцієнтами $a_{10} = a_{N,N+1} = 0$):

$$\beta_m x_{m-1} + \gamma_m x_m + \delta_m x_{m+1} = b_m; \quad m = 1,2,...,N. \tag{5.42}$$

Згідно зі схемою *правої прогонки* розв'язок СЛАР знаходиться у вигляді

$$x_m = \mu_{m+1} x_{m+1} + \eta_{m+1}; \quad m = N-1,...,1. \tag{5.43}$$

Для визначення коефіцієнтів це рівняння записують для компоненти $x_{m-1} = \mu_m x_m + \eta_m$ та обидва підставляють у (5.42). Отримане рівняння

$$[(\beta_m \mu_m + \psi_m)\mu_{m+1} + \delta_m]x_{m+1} + [(\beta_m \mu_m + \gamma_m)\eta_{m+1} + \beta_m \eta_m - b_m] = 0 \tag{5.44}$$

буде справедливим для будь-яких x_{m+1}, якщо вирази у квадратних дужках завжди дорівнюють нулю. Тому коефіцієнти

$$\mu_{m+1} = -\delta_m / (\beta_m \mu_m + \gamma_m); \quad \eta_{m+1} = (b_m - \beta_m \eta_m)/(\beta_m \mu_m + \gamma_m); \quad m = 1,2,...,N-1. \tag{5.45}$$

$$\mu_1 = -a_{12}/a_{11}; \quad \eta_1 = b_1/a_{11}. \tag{5.46}$$

Для визначення x_N використовується система рівнянь $a_{N,N-1}x_{N-1} + a_{NN}x_N = b_N$ та $x_{N-1} = \mu_N x_N + \eta_N$, що призводить до розв'язку

$$x_N = (a_{NN}\eta_N - b_N)/(a_{N,N-1}\mu_N + a_{NN}). \quad (5.47)$$

Спочатку застосовуються формули (5.46) і (5.45) прямої прогонки. Потім – формули (5.47) і (5.43) зворотної прогонки.

Згідно зі схемою *лівої прогонки* розв'язок СЛАР знаходиться у вигляді

$$x_{m+1} = \xi_{m+1}x_m + \lambda_{m+1}; \quad m = 1,..., N-1. \quad (5.48)$$

Аналогічно попередньому можна отримати:

$$\xi_m = -\beta_m/(\delta_m\xi_{m+1} + \gamma_m); \quad \lambda_m = (b_m - \delta_m\lambda_{m+1})/(\delta_m\xi_{m+1} + \gamma_m); \quad m = N-1,...,1. \quad (5.49)$$

$$\xi_N = -a_{N,N-1}/a_{NN}; \quad \lambda_N = b_N/a_{NN}. \quad (5.50)$$

Для визначення x_1 використовується система рівнянь $a_{11}x_1 + a_{12}x_2 = b_1$ та $x_2 = \xi_2 x_1 + \lambda_2$, що призводить до розв'язку

$$x_1 = (b_1 - a_{12}\lambda_2)/(a_{12}\xi_2 + a_{11}). \quad (5.51)$$

Послідовність обчислення формул аналогічна: (5.50) і (5.49) – пряма прогонка, (5.51) і (5.48) – зворотна.

Є ще схема *зустрічної* прогонки, яка буває корисна, якщо необхідно знайти значення лише для одного x_j при $1 < j < N$. Організовуються прогонки з двох кінців (зустрічні) до цього j. Формули для визначення коефіцієнтів та невідомих – ті ж самі, але є додаткова формула "зшивання":

$$x_j = (\eta_{j+1} + \mu_{j+1}\lambda_{j+1})/(1 - \mu_{j+1}\lambda_{j+1}). \quad (5.52)$$

Доведено, що достатні умови, при виконанні яких формули (5.45), (5.47), (5.49), (5.51) і (5.52) мають сенс, є такими:

$$|\gamma_m| \geq |\beta_m| + |\delta_m|; \quad |\mu_1| \leq 1; \quad |\xi_N| \leq 1; \quad |\mu_1| + |\xi_N| < 2. \quad (5.53)$$

Крім того, доведено, що при проведення обчислень на ЕОМ похибки округлення дають сумарну похибку, пропорційну $N^2\varepsilon_M$, де ε_M – машинний іпсилон (див. Таблицю 1.5. Розділу 1).

Контрольні питання до підрозділу 5.1

1. На які дві великі групи ділять методи розв'язування СЛАР?
2. Чи можливо приведення матриці СЛАР до симетричного стану?

Контрольні питання до підрозділу 5.2

1. Як обчислити число обумовленості матриці?
2. Про що свідчить завелике значення числа обумовленості.

Контрольні питання до підрозділу 5.3

1. Чому формули Крамера не застосовують при кількості невідомих $N > 3$?
2. Що відбувається на прямому та зворотному ходах методу Гаусса? Які недоліки цього методу?
3. Яку (приблизно) кількість операцій мають прямі методи розв'язування СЛАР?
4. За рахунок чого схема Холецького має перевагу над класичним методом Гаусса?
5. Які спільні риси мають всі прямі методи розв'язування СЛАР?
6. В яких випадках стає в нагоді ортогоналізація Грама-Шмідта?

Контрольні питання до підрозділу 5.4

1. Чому для СЛАР з тридіагональною матрицею розроблено особливі алгоритми?

Розділ 6

ОБЧИСЛЕННЯ ВЛАСНИХ ЗНАЧЕНЬ І ВЕКТОРІВ МАТРИЦЬ

6.1. Загальні зауваження

Власними значеннями (*характеристичними числами*) квадратної матриці $[A]$ називають ті значення скалярного параметра λ, для яких матриця $[A]-\lambda[I]$ є виродженою. *Спектром* матриці $[A]$ називається вся множина її власних значень. Він співпадає з множиною коренів алгебраїчного рівняння

$$\det([A]-\lambda\cdot[I])\equiv D(\lambda)\equiv \det|\,a_{mn}-\lambda\cdot\delta_{mn}\,|=0\,, \qquad (6.1)$$

яке при цьому називається характеристичним рівнянням. Серед власних значень є окремі та можуть бути й кратні власні значення. Зазвичай всі власні значення нумерують за принципом збільшення: $0\le|\lambda_1|\le|\lambda_2|\le\ldots\le|\lambda_N|$.

Деякі важливі властивості:

- спектральний радіус $s(A)\le\|[A]\|$ для несиметричної та $\max s(A)=\|[A]\|$ для симетричної матриці, коли ці норми – одна з m, k чи w (див. підрозділ 4.2);
- *подібні* матриці мають однаковий спектр;
- якщо матриця $[A]$ позитивно визначена, то всі її власні значення – дійсні;
- якщо матриця $[A]$ з дійсними елементами a_{mn} – *симетрична*, то:
  - всі її N власних значень – дійсні;
  - існують N взаємно ортогональних *власних векторів* матриці $\{x\}_j$, які створюють базис простору E^N розмірністю N, причому

$$[A]\{x\}_j=\lambda_j\{x\}_j;\quad j=1,2,\ldots,N\,. \qquad (6.2)$$

Є чимало методів знаходження власних значень і власних векторів матриці. Їх можна розділити на методи для знаходження:

- всіх власних значень і власних векторів;
- границь спектра власних значень;
- декількох власних значень і власних векторів.

Інша класифікація методів така:

- прямі методи розгортання характеристичного рівняння (6.1);
- прямі методи приведення матриці $[A]$ до діагонального вигляду шляхом застосування перетворень подібності;
- ітераційні методи для отримання декількох власних характеристик.

6.2. Обчислення всіх власних значень квадратної матриці

Є декілька класичних методів знаходження всіх власних значень. Вони використовують представлення характеристичного рівняння (6.1) у вигляді алгебраїчного полінома з невідомими коефіцієнтами, після визначення яких застосовується

один з методів знаходження його коренів (див. Розділ 3). Опишемо лише два методи: прямого розгортання характеристичного рівняння (самий повільний), та метод Данилевського (самий швидкий). Є ще методи Крилова, Леверр'є, невизначених коефіцієнтів, інтерполювання, повороту, інші.

6.2.1. Метод прямого розгортання характеристичного рівняння

Характеристичне рівняння (6.1) представляється у вигляді:

$$D(\lambda) = (-1)^N [\lambda^N - \sigma_1 \lambda^{N-1} + \sigma_2 \lambda^{N-2} - ... + (-1)^N \sigma_N], \tag{6.3}$$

де σ_j є суми всіх діагональних мінорів j-го порядку матриці $[A]$, зокрема, $\sigma_N = \det[A]$. Можна підрахувати, що тільки для обчислення всіх коефіцієнтів σ_j полінома (6.3) потрібно $2^N - 1$ разів обчислити детермінанти матриці всіх порядків (від 1 до N), що є фактично нереальним при значних величинах N.

6.2.2. Метод Данилевського

Якщо привести характеристичне рівняння (6.1) до *нормального вигляду Фробеніуса*

$$D(\lambda) = \begin{vmatrix} p_1 - \lambda; & p_2; & p_3; & ... & p_N \\ 1; & -\lambda; & 0; & ... & 0 \\ 0; & 1; & -\lambda; & ... & 0 \\ ... & ... & ... & ... & ... \\ 0; & 0; & 0; & ... & -\lambda \end{vmatrix}, \tag{6.4}$$

то потім можна легко отримати алгебраїчний поліном N-ої степені:

$$D(\lambda) = (-1)^N \cdot (\lambda^N - p_1 \lambda^{N-1} - p_2 \lambda^{N-2} - ... - p_N). \tag{6.5}$$

Матрицю Фробеніуса

$$[\Phi] = \begin{bmatrix} p_1; & p_2; & ... & p_{N-1}; & p_N \\ 1; & 0; & ... & 0; & 0 \\ 0; & 1; & ... & 0; & 0 \\ ... & ... & ... & ... & ... \\ 0; & 0; & ... & 1; & 0 \end{bmatrix} \tag{6.6}$$

можна зв'язати з матрицею $[A]$ співвідношенням

$$[\Phi] = [S]^{-1}[A][S], \tag{6.7}$$

де матриця $[S]$ – якась не вироджена матриця. Згідно з останнім виразом матриці $[A]$ та $[\Phi]$ є подібними (див. підрозділ 4.1), тому мають однаковий набір власних значень і замість полінома (6.3) можна використовувати поліном (6.5).

У методі Данилевського перехід від матриці $[A]$ до матриці $[\Phi]$ проводиться за $N-1$ кроків перетворень. Для кожного k-го рядка матриці $[A]$, починаючі з останнього, обчислюються:

$$b_{mn} = a_{mn} + a_{m,k-1} w_{k-1,n} \quad \text{при} \quad 1 \le m \le k; \quad n \ne k-1; \tag{6.8}$$

$$b_{m,k-1} = a_{m,k-1} w_{k-1,k-1} \quad \text{при} \quad 1 \le m \le k, \text{ де} \tag{6.9}$$

$$w_{k-1,n} = -a_{kn} / a_{k,k-1} \quad \text{при} \quad n \neq k-1 \quad \text{та} \quad w_{k-1,k-1} = 1/a_{k,k-1}, \tag{6.10}$$

а b_{mn} є компонентами проміжної матриці $[B]$. Для завершення перетворення ще необхідно перейти до матриці $[C]$ з компонентами

$$c_{mn} = b_{mn} \quad \text{при} \quad 1 \le m \le k-2; \quad c_{k-1,n} = \sum_{j=1}^{k} a_{k,j} b_{j,n} \quad \text{при} \quad 1 \le n \le k. \tag{6.11}$$

Матриця $[C]$ має перетворений k-й рядок у вигляді всіх нулів, окрім одиниці на діагоналі.

Всі операції повторюються для нового k-го рядка, причому замість матриці $[A]$ використовується матриця $[C]$. Проблеми виникають, коли компонента $a_{k,k-1}$, на яку потрібно ділити, дорівнює нулю. Можливі два випадки.

У першому випадку в k-му рядку *ліворуч* компоненти $a_{k,k-1}$ є хоча би одна не нульова компонента a_{kj}, $j < k-1$. Тоді достатньо переставити місцями $(k-1)$-й та j-й стовпці та такі ж рядки матриці, оскільки доведено, що нова матриця буде подібною до старої.

У другому випадку в k-му рядку *ліворуч* компоненти $a_{k,k-1}$ немає не нульових компонент. Це означає, що в матриці ліворуч та нижче компоненти $a_{k,k-1}$ всі компоненти – нульові, або, інакше кажучи, що така проміжна матриця $[A]$ містить 4 блоки, причому той, що зліва внизу – нульовий: $[A] = \begin{bmatrix} [A]_1 & [L] \\ [0] & [A]_2 \end{bmatrix}$.

Тоді її характеристичне рівняння розпадається на два:

$$\det([A] - \lambda[I]) = \det([A]_1 - \lambda[I]) \cdot \det([A]_2 - \lambda[I]), \tag{6.12}$$

причому матриця $[A]_2$ вже приведена до канонічної форми Фробеніуса, залишилося зробити це з матрицею $[A]_1$.

Якщо $\{z\} = \{z_1, z_2, ..., z_N\}^T$ – власний вектор матриці [Ф], то він задовольняє системі $[\Phi]\{z\} = \lambda\{z\}$ або $([\Phi] - \lambda[I])\{z\} = \{0\}$. Це й є система для обчислення компонент власного вектора $\{z\}$. Оскільки ця система – однорідна, то з неї можливо знайти розв'язок лише з точністю до деякого множника. Якщо прийняти, що $z_N = 1$, то послідовно отримаємо, що

$$z_{N-1} = \lambda; \quad z_{N-2} = \lambda^2; \quad ..., \quad z_1 = \lambda^{N-1}. \tag{6.13}$$

Але нам потрібно знайти $\{x\} = \{x_1, x_2, ..., x_N\}^T$ – власний вектор матриці $[A]$. Якщо з компонент w_{mn} (див. формули (6.10)) створити матриці

$$[W]_{k-1} = \begin{bmatrix} 1; & 0; & ... & 0; & ... & 0 \\ 0; & 1; & ... & 0; & ... & 0 \\ ... & ... & ... & ... & ... & ... \\ w_{k-1,1}; & w_{k-1,2}; & ... & w_{k-1,k-1}; & ... & w_{k-1,N} \\ ... & ... & ... & ... & ... & ... \\ 0; & 0; & ... & 0; & ... & 1 \end{bmatrix}, \tag{6.14}$$

то можна одержати, що

$$\{x\} = [W]_{N-1} \cdot [W]_{N-2} \cdot \ldots \cdot [W]_2 \cdot [W]_1 \{z\}, \tag{6.15}$$

причому кожна матриця $[W]_j$ при перемноженні модифікує лише одне j-е значення вектора. Цей процес необхідно повторювати для кожного значення λ.

Отже, метод Данилевського дозволяє знаходити не тільки характеристичне рівняння матриці, а й власні вектори (після знаходження власних значень).

6.3. Обчислення границь спектра власних значень

Досить часто достатньо знати тільки границі спектра власних значень (див. Розділ 7).

Позначимо як $\underline{\lambda}(A)$ та $\overline{\lambda}(A)$ – відповідно мінімальне та максимальне власні значення матриці $[A]$. Для наближеного обчислення границь спектра власних значень матриці зазвичай використовують теорему Гершгоріна та метод Люстерніка. Теорема Гершгоріна стверджує, що всі власні значення $\lambda_k(A)$ матриці $[A]$ (дійсні та комплексні) належать об'єднанню кіл

$$| z - a_{mm} | \le s_m; \quad m = 1, 2, \ldots, N, \tag{6.16}$$

де спектральний радіус m–го кола

$$s_m = \sum_{\substack{n=1 \\ n \ne m}}^{N} | a_{mn} |. \tag{6.17}$$

Одним з наслідків теореми Гершгоріна є оцінка для спектрального радіуса матриці (максимального за модулем власного значення):

$$s(A) = | \overline{\lambda}(A) | \le \max_m \sum_{n=1}^{N} | a_{mn} |. \tag{6.18}$$

Якщо ввести допоміжну матрицю $[B] = \overline{\lambda}(A) \cdot [I] - [A]$, то можна спочатку отримати $| \overline{\lambda}(B) | \le \max_m \sum_{n=1}^{N} | b_{mn} |$, а потім:

$$\underline{\lambda}(A) = \overline{\lambda}(A) - \overline{\lambda}(B), \tag{6.19}$$

причому ця оцінка може бути дуже приблизною, навіть мати невірний знак, оскільки $\overline{\lambda}(A)$ і $\overline{\lambda}(B)$ – близькі за величинами великі числа.

Згідно з методом Люстерніка в ітераційному процесі спочатку обчислюється максимальне власне значення $\overline{\lambda}(A)$:

$$\overline{\lambda}(A) = \lim_{k \to \infty} \| \{\varphi\}^{(k)} \|, \quad \text{де} \quad \{\varphi\}^{(k)} = [A] \frac{\{\varphi\}^{(k-1)}}{\| \{\varphi\}^{(k-1)} \|}; \quad k = 1, 2, \ldots, \tag{6.20}$$

причому

$$\{x\}_1 = \lim_{k \to \infty} \{\varphi\}^{(k)}; \quad k = 1, 2, \ldots. \tag{6.21}$$

Цей метод збігається й тоді, коли $\overline{\lambda}(A)$ є кратним.

Потім, якщо це потрібно, можна обчислити мінімальне власне значення $\underline{\lambda}(A)$. Знову вводиться допоміжна матриця $[B] = \overline{\lambda}(A) \cdot [I] - [A]$, в ітераціях знаходиться:

$$\underline{\lambda}(A)=\lim_{k\to\infty}\frac{\|[A]\{\psi\}^{(k)}\|}{\|\{\psi\}^{(k)}\|},\ \text{де}\ \ \{\psi\}^{(k)}=[B]\frac{\{\psi\}^{(k-1)}}{\|\{\psi\}^{(k-1)}\|};\ \ k=1,2,\ldots. \tag{6.22}$$

Початкові вектори $\{\varphi\}^{(0)}$ та $\{\psi\}^{(0)}$ можуть бути довільними, але існують такі, при яких процес зовсім не збігається. Тоді потрібно обрати інший початковий вектор і провести всі обчислення знову.

Ітерації закінчуються, коли значення $\overline{\lambda}(A)$ та $\underline{\lambda}(A)$ стабілізуються з призначеною точністю. В одній ітерації обчислення (6.20) та (6.22), при повністю заповнених матрицях, потрібно біля N^2 операцій, тобто відносно небагато, але швидкість збіжності цих процесів не є значною.

Примітка 6.1. Іноді для визначення $\overline{\lambda}(A)$ більш швидким буде дещо інший ітераційний процес, ніж (6.20), (6.21):

$$\overline{\lambda}(A)=\lim_{k\to\infty}\frac{(\{\varphi\}^{(k)},\{\widetilde{\varphi}\}^{(k)})}{(\{\varphi\}^{(k-1)},\{\widetilde{\varphi}\}^{(k)})};\quad k=1,2,\ldots, \tag{6.23}$$

де

$$\{\varphi\}^{(k)}=[A]\{\varphi\}^{(k-1)}=[A]^k\{\varphi\}^{(0)};\quad \{\widetilde{\varphi}\}^{(k)}=[A]^T\{\widetilde{\varphi}\}^{(k-1)}=([A]^T)^k\{\varphi\}^{(0)};\quad k=1,2,\ldots. \tag{6.24}$$

Цей процес особливо зручний, коли матриця $[A]$ є симетричною, оскільки тоді $[A]^T=[A]$, $\{\widetilde{\varphi}\}^{(k)}=\{\varphi\}^{(k)}$ і не потрібно будувати окремий процес згідно з другою формулою (6.24). Його називають *методом скалярних добутків*.

Примітка 6.2. Для прискорення збіжності ітераційних процесів (з простими (повільними) законами збіжності) можна застосовувати наступну формулу:

$$\overline{\lambda}(A)\approx\frac{(\{\varphi\}^{(k+1)},\{\varphi\}^{(k-1)})-(\{\varphi\}^{(k)},\{\varphi\}^{(k)})}{\|\{\varphi\}^{(k+1)}-2\{\varphi\}^{(k)}+\{\varphi\}^{(k-1)}\|}. \tag{6.25}$$

Примітка 6.3. Для визначення норм матриць $[A]$ і $[A]^{-1}$ можна використовувати формули

$$\|[A]\|=\sqrt{\overline{\lambda}(A^*A)}\quad\text{та}\quad\|[A]^{-1}\|=1/\sqrt{\underline{\lambda}(A^*A)}, \tag{6.26}$$

де $[A]^*$ – матриця, спряжена з матрицею $[A]$.

6.4. Обчислення декількох власних значень квадратної матриці

При розв'язуванні крайових задач динаміки, досить часто потрібні не всі власні значення, а лише декілька найменших власних значень або деякий діапазон значень. Для їхнього знаходження розроблено декілька ітераційних методів: послідовної ортогоналізації, ітерацій, вичерпування, інші. Власно кажучи, цими методами можна послідовно знайти всі власні характеристики, але це не буде кращим варіантом. У цьому Розділі розглянемо лише зазначені методи, інші – в Розділі 26.

6.4.1. Метод послідовної ортогоналізації

Вважається, що матриця $[A]$ є симетричною.

Спочатку, згідно з ітераційним процесом (6.20), знаходиться найбільше власне значення та відповідний йому власний вектор, позначимо їх як λ_1 та

$\{x\}_1$. Для знаходження наступних λ_2 та $\{x\}_2$ проводиться так званий "зсув спектра" шляхом введення нової матриці

$$[B]=[A]-\lambda_1[I], \tag{6.27}$$

а також виключення компонент власного вектора $\{x\}_1$ за формулою

$$\{\tilde{\psi}\}^{(k)}=\{\psi\}^{(k)}-\alpha_1\cdot\{x\}_1;\quad k=1,2,\ldots. \tag{6.28}$$

Параметр α_1 обчислюється в ітераціях з умови ортогональності поточного вектора $\{\tilde{\psi}\}^{(k)}$ власному вектору $\{x\}_1$, яка створюється із застосуванням (6.28) як скалярний добуток, який дорівнює нулю:

$$(\{\tilde{\psi}\}^{(k)},\{x\}_1)=(\{\psi\}^{(k)},\{x\}_1)-\alpha_1\cdot(\{x\}_1,\{x\}_1)=0;\quad k=1,2,\ldots. \tag{6.29}$$

З (6.29)

$$\alpha_1=\lim_{k\to\infty}\frac{(\{\psi\}^{(k)},\{x\}_1)}{(\{x\}_1,\{x\}_1)};\quad k=1,2,\ldots. \tag{6.30}$$

Початковий вектор $\{\psi\}^{(0)}$ може бути довільним, але відмінним від $\{x\}_1$, а всі наступні обчислюються як

$$\{\psi\}^{(k+1)}=-[B]\frac{\{\tilde{\psi}\}^{(k)}}{\|\{\tilde{\psi}\}^{(k)}\|};\quad k=1,2,\ldots. \tag{6.31}$$

Внаслідок ітераційного процесу знаходиться

$$\lambda_2(A)=\lim_{k\to\infty}\frac{\|[A]\{\tilde{\psi}\}^{(k)}\|}{\|\{\tilde{\psi}\}^{(k)}\|};\quad k=1,2,\ldots. \tag{6.32}$$

Цей алгоритм узагальнюється на випадок визначення деякого (n)-го власного значення та вектора. Будемо вважати, що попередні $(n-1)$ значень та векторів знайдено. Тоді замість (6.27), (6.28) і (6.30) відповідно:

$$[B]=[A]-\lambda_{n-1}[I]; \tag{6.33}$$

$$\{\tilde{\psi}\}^{(k)}=\{\psi\}^{(k)}-\sum_{j=1}^{n-1}\alpha_j\cdot\{x\}_j;\quad k=1,2,\ldots; \tag{6.34}$$

$$\alpha_j=\frac{(\{\psi\}^{(k)},\{x\}_j)}{(\{x\}_j,\{x\}_j)};\quad k=1,2,\ldots;\quad j=1,\ldots,n-1. \tag{6.35}$$

Внаслідок ітераційного процесу знаходиться

$$\lambda_n(A)=\lim_{k\to\infty}\frac{\|[A]\{\tilde{\psi}\}^{(k)}\|}{\|\{\tilde{\psi}\}^{(k)}\|};\quad k=1,2,\ldots, \tag{6.36}$$

а також

$$\{x\}_n=\lim_{k\to\infty}\{\tilde{\psi}\}^{(k)};\quad k=1,2,\ldots. \tag{6.37}$$

Якщо матриця $[A]$ не є симетричною, то алгоритм майже тотожний, але паралельно проводяться два процеси: з матрицями $[A]$ та $[A^*]$ для систем $[A]\{u\}=\lambda\cdot\{u\}$ та $[A^*]\{u^*\}=\lambda^*\cdot\{u^*\}$. Тільки замість (6.30) і (6.35) використовуються формули

$$\alpha_j=\frac{(\{\psi\}^{(k)},\{x^*\}_j)}{(\{x\}_j,\{x^*\}_j)};\quad \alpha_j^*=\frac{(\{\psi^*\}^{(k)},\{x\}_j)}{(\{x\}_j,\{x^*\}_j)};\quad k=1,2,\ldots;\quad j=1,\ldots,n-1, \tag{6.38}$$

які утворюються із умов "перехресної" ортогональності $(\{\widetilde{\psi}\}^{(k)},\{x^*\}_j)=0$ і $(\{\widetilde{\psi}^*\}^{(k)},\{x\}_j)=0$ відповідно. Внаслідок такої ортогоналізації власні значення λ_n та λ_n^*, а також власні вектори $\{x\}_n$ та $\{x^*\}_n$ співпадають.

Потрібно додатково вказати, що наявність похибок округлення при обчисленні в ЕОМ призводить до нестійкого процесу. Тому рекомендують проводити ортогоналізацію не на кожному кроці $k=1,2,\ldots$, а через порівняно значну їхню кількість.

6.4.2. Степеневі методи (метод ітерацій і метод зворотних ітерацій)

У методі ітерацій (Power Method – PM) послідовно обчислюються вектори

$$\{x\}^{(k+1)}=\alpha_{(k)}\cdot[A]\{x\}^{(k)};\quad k=1,2,\ldots, \tag{6.39}$$

починаючи з довільного вектора $\{x\}^{(1)}$, хоча рекомендують його приймати у вигляді нульового з однією компонентою, яка дорівнює одиниці, або наближеного до власного вектора $\{x\}_1$. Ця послідовність збігається до деякого вектора $\{x\}$ такого, що максимальне власне значення $\lambda_1=\overline{\lambda}(A)$ обчислюється як

$$\lambda_1=\lim_{k\to\infty}\frac{(\{x\}^{(k)},[A]\{x\}^{(k)})}{(\{x\}^{(k)},\{x\}^{(k)})};\quad k=1,2,\ldots. \tag{6.40}$$

Якщо множник $\alpha_{(k)}$ обирати таким чином, щоб максимальна за модулем компонента вектора $\{x\}^{(k+1)}$ дорівнювала одиниці, то цей вектор збігається до власного вектора $\{x\}_1$. Якщо прийняти $\alpha_{(k)}\equiv 1$, то для отримання власного вектора після стабілізації процесу (6.39) потрібно провести нормування отриманого вектора. Метод збігається повільно, причому тим повільніше, чим ближче до λ_n є сусіднє власне значення. Для прискорення збіжності, як це можна легко збагнути з формули (6.39), можна замість матриці $[A]$ використовувати матриці $[A]^2$, $[A]^3$, … .

Для знаходження наступних λ_n та $\{x\}_n$, $n=2,\ldots$ проводиться "зсув спектра" (див. формули (6.27) ... (6.30), (6.33) ... (6.35)), причому у формулах (6.39), (6.40) замість матриці $[A]$ вже буде застосовуватися матриця $[B]$.

Якщо деяке з власних значень λ_n має кратність $m>1$, то для знаходження інших власних векторів потрібно обирати інший початковий вектор і таким чином врешті-решт знайти потрібну кількість $m>1$ власних векторів з кратним власним значенням λ_n.

У методі *зворотних* ітерацій (Inverse Power Method – Inverse PM) багатократно розв'язується СЛАР

$$\alpha_{(k)}\cdot[B]\{x\}^{(k+1)}=\{x\}^{(k)};\quad k=1,2,\ldots, \tag{6.41}$$

причому вибір початкового вектора та $\alpha_{(k)}$ є такими, як й для методу ітерацій, а матриця $[B]$ один раз зазнає розклад Холецького: $[B]=[L][U]$ або $[B]=[U]^T[D][U]$ у випадку симетричності матриці $[B]=[A]-\sigma[I]$. Більша популярність цього методу

(ніж РМ) пов'язана з тим, що швидкість збіжності методу значно підвищується, коли обрано σ, близьке до знайденого власного значення λ_n (при $\sigma = \lambda_n$ достатньо однієї ітерації). Є й недолік: власні вектори, обчислені для двох дуже близьких власних значень, не зовсім ортогональні через похибки округлення та зупинення ітераційного процесу (критерій збіжності – стабілізація $\rho = 1/\| \{x\}^{(k+1)} \|$). Зазвичай їх додатково ортогоналізують відносно раніше знайдених власних векторів. Якщо в ітераціях у вірному напрямку змінювати величину "зсуву спектра", то швидкість збіжності можна значно підвищити.

Примітка 6.4. Кількість від'ємних значень компонент діагональної матриці $[D]$ розкладу $[B] = [U]^T [D][U]$, де симетрична матриця $[B] = [A] - \sigma[I]$, вказує на кількість власних значень матриці $[A]$, які менше власного значення $\lambda_n = \sigma$. Це може застосовуватися для перевірки: всі або ні власні значення були знайдені у обраному діапазоні, а також для приблизного визначення власних значень (знаходяться діапазони з одним власним значенням).

6.4.3. Методи вичерпування

Методами вичерпування традиційно називають всі методи, які для знаходження наступних власних значень і векторів виключають з розгляду вже непотрібні для цього частини системи, тим самим прискорюючи процес.

У варіанті вичерпування *відніманням* для знаходження наступних власних значень і векторів "зсув спектра" проводиться дещо іншим способом, ніж у формулах (6.27), (6.28), (6.33), (6.34). Вводиться нова матриця (прийнято, що $[A]_1 = [A]$):

$$[A]_{n+1} = [A]_n - \lambda_n \cdot \{x\}_n \cdot (\{\tilde{x}\}_n)^T; \quad n = 1, 2, ..., N-1, \tag{6.42}$$

де вектор $\{\tilde{x}\}_n$ отримують з власного вектора $\{x\}_n$ шляхом нормування його таким чином, щоб $(\{x\}_n, \{\tilde{x}\}_n) = 1$. Доведено, що нова матриця має ті ж самі власні вектори, що й вихідна матриця $[A]$, а всі раніше отримані власні значення $\lambda_1, ..., \lambda_n$ дорівнюють нулю, тобто перше власне значення матриці $[A]_{n+1}$ є λ_{n+1} матриці $[A]$. Його визначають за будь-яким методом. При цьому для знаходження послідовності векторів можна використовувати таку властивість матриць $[A]_{n+1}$:

$$([A]_{n+1})^k \{\varphi\}^{(0)} = ([A]_n)^k \{\varphi\}^{(0)} - \lambda_n \cdot \{x\}_n \cdot (\{\tilde{x}\}_n)^T \{\varphi\}^{(0)}, \tag{6.43}$$

яка дозволяє уникнути ітерацій з безпосереднім застосуванням нових матриць.

Є ще методи вичерпування *шляхом звуження* та *подібних перетворень*.

В останньому методі кожна нова матриця має на одиницю менший розмір. Якщо взяти довільну *ортогональну* матрицю $[P]$, першим стовпцем якої є власний вектор $\{x\}_1$, то, оскільки $[P]^T = [P]^{-1}$ та $[P]^T \{x\}_1 = \{e\}_1 = \{1, 0, ..., 0\}^T$:

$$[P]^T [A][P] = \begin{bmatrix} \lambda_1 & [0] \\ [0] & [\tilde{A}] \end{bmatrix}, \tag{6.44}$$

тобто утворилася нова матриця $[\tilde{A}]$ на одиницю меншого розміру, причому $\lambda_2 = \overline{\lambda}(\tilde{A})$.

Контрольні питання до підрозділу 6.1

1. Які математичні об’єкти називають власними значеннями (характеристичними числами) квадратної матриці?

2. Наведіть загальну класифікацію методів знаходження власних значень квадратної матриці.

Контрольні питання до підрозділу 6.2

1. Яке характерне заповнення має матриця Фробеніуса?

2. За скільки кроків перетворень у методі Данилевського відбувається перехід від матриці $[A]$ до матриці Фробеніуса?

Контрольні питання до підрозділу 6.3

1. Якій проблемі присвячений метод Люстерніка?

Контрольні питання до підрозділу 6.4

1. Яке є обмеження для застосування методу послідовної ортогоналізації? Який суттєвий недолік має цей метод?

2. Яку швидкість збігання розв’язку мають степеневі методи?

3. Як можна визначити, чи всі власні значення були знайдені у обраному діапазоні?

4. Які методи називають методами вичерпування?

5. Для чого застосовують "зсув спектра"?

Розділ 7

ІТЕРАЦІЙНІ МЕТОДИ РОЗВ'ЯЗУВАННЯ СИСТЕМ ЛІНІЙНИХ АЛГЕБРАЇЧНИХ РІВНЯНЬ

У цьому Розділі будуть розглянуті основні, в більшості випадків класичні, методи розв'язування лінійних СЛАР, оскільки ця тема дуже велика.

В ітераційних методах розв'язування СЛАР (5.1), тобто

$$[A]\{x\} = \{b\}, \tag{7.1}$$

вектор-розв'язок $\{x\}^{(k)}$ змінюється в ітераціях. Вважається, що алгоритм збігається до точного вектора-розв'язку $\{x\}$, якщо норма похибки $\| \{x\}^{(k)} - \{x\} \| \to 0$ при $k \to \infty$, де $k = 0,1,...$ – номер ітерації. Ясна річ, що нескінченну кількість ітерацій проводити немає сенсу хоча б тому, що "обрізання" дійсних чисел в ЕОМ ніколи не дозволить одержати точний розв'язок, який, до того ж, зазвичай невідомий. Тому необхідно призначати розумну величину похибки розв'язку $\varepsilon > 0$ та обмежувати ітерації однією з умов:

$$\| \{x\}^{(k+1)} - \{x\}^{(k)} \| \le \varepsilon \cdot \| \{x\}^{(k+1)} \| \quad \text{або} \quad \| \{r\}^{(k+1)} - \{r\}^{(k)} \| \le \varepsilon \cdot \| \{r\}^{(k)} \| \tag{7.2}$$

(це класичні умови), а краще такою, що відображає умову зниження похибки наближення на вказану кількість порядків p, тобто при призначенні $\varepsilon = 10^{-p}$:

$$\| \{r\}^{(k+1)} \| \le \varepsilon \cdot \| \{r\}^{(0)} \| \quad \text{або} \quad \left| (\{x\}^{(k+1)})^T \{r\}^{(k+1)} \right| \le \varepsilon^2 \cdot \left| (\{x\}^{(0)})^T \{r\}^{(0)} \right|, \tag{7.3}$$

де $\{r\}^{(k)}$ – вектор похибок наближення рівняння $[A]\{x\} = \{b\}$:

$$\{r\}^{(k)} = \{b\} - [A]\{x\}^{(k)}, \quad k = 0,1,... . \tag{7.4}$$

Для гарантованого припинення ітерацій застосовують, зокрема, поступове зниження необхідної точності при перевищенні кількості ітерацій деякого визначеного значення, зазвичай, відносно кількості невідомих у СЛАР.

7.1. Ітераційні методи розв'язування СЛАР, які використовують спектральні характеристики матриці

Найчастіше ітераційні схеми розв'язування СЛАР будують двошаровими. Але застосовують і тришарові схеми, які можуть розглядатися як засіб прискорення двошарових схем (див. підрозділ 7.4).

7.1.1. Еквівалентні форми двошарових ітераційних схем

Будь-яка лінійна двошарова ітераційна схема для розв'язування СЛАР (7.1) $[A]\{x\} = \{b\}$ може бути представлена як

$$[Q]^{(k)} \{x\}^{(k+1)} = [R]^{(k)} \{x\}^{(k)} + \{f\}; \quad k = 0,1,..., \tag{7.5}$$

де матриці $[Q]$ і $[R]$ можуть змінюватися в ітераціях. Матриця $[Q]$ називається *матрицею розщеплення*, повинна мати обернену матрицю $[Q]^{-1}$.

Точний вектор-розв'язок $\{x\}$ задовольняє рівнянню (7.5), тобто необхідно, щоб $([Q]^{(k)}-[R]^{(k)})\{x\}=\{f\}$ для всіх $k=0,1,\ldots$. Із (7.1) $\{x\}=[A]^{-1}\{b\}$, тому $([Q]^{(k)}-[R]^{(k)})[A]^{-1}\{b\}\equiv\{f\}$. Отже, існує матриця $([Q]^{(k)}-[R]^{(k)})^{-1}$ така, що $\{b\}\equiv[A]([Q]^{(k)}-[R]^{(k)})^{-1}\{f\}$. Завжди можна прийняти, що

$$([Q]^{(k)}-[R]^{(k)})/\tau^{(k+1)}=[A],\quad \text{тоді}\quad \{f\}=\tau^{(k+1)}\{b\};\quad k=0,1,\ldots. \tag{7.6}$$

Остаточно двошарова ітераційна схема (7.5) записується в *канонічній* формі:

$$\boxed{[Q]^{(k)}\frac{\{x\}^{(k+1)}-\{x\}^{(k)}}{\tau^{(k+1)}}+[A]\{x\}^{(k)}=\{b\}};\quad k=0,1,\ldots. \tag{7.7}$$

У книзі [77] ця схема називається *екстрапольованою*, представляється як

$$\{x\}^{(k+1)}=\tau^{(k+1)}\cdot([\overline{G}]^{(k)}\{x\}^{(k)}+\{\overline{\beta}\}^{(k)})+(1-\tau^{(k+1)})\{x\}^{(k)};\quad k=0,1,\ldots, \tag{7.8}$$

де введені позначення

$$[\overline{G}]^{(k)}=[I]-([Q]^{(k)})^{-1}[A];\quad \{\overline{\beta}\}^{(k)}=([Q]^{(k)})^{-1}\{b\}. \tag{7.9}$$

Якщо застосувати (7.4) та ввести вектор поправки

$$\{w\}^{(k)}=([Q]^{(k)})^{-1}\{r\}^{(k)}=([Q]^{(k)})^{-1}(\{b\}-[A]\{x\}^{(k)}), \tag{7.10}$$

то (7.7) отримає вигляд *рекурентної* схеми:

$$\{x\}^{(k+1)}=\{x\}^{(k)}+\tau^{(k+1)}\{w\}^{(k)};\quad k=0,1,\ldots. \tag{7.11}$$

Вираз (7.7) ще можна переписати таким чином:

$$\{x\}^{(k+1)}=[G]^{(k+1)}\{x\}^{(k)}+\{\beta\}^{(k+1)};\quad k=0,1,\ldots, \tag{7.12}$$

де введені позначення

$$[G]^{(k+1)}=[I]-\tau^{(k+1)}([Q]^{(k)})^{-1}[A];\quad \{\beta\}^{(k+1)}=\tau^{(k+1)}([Q]^{(k)})^{-1}\{b\}. \tag{7.13}$$

Матрицю $[G]$ називають матрицею переходу.

Отже, є декілька *еквівалентних* форм (формули (7.7), (7.8), (7.11) і (7.12)) представлення двошарової ітераційної схеми для розв'язування СЛАР (7.1).

Якщо матриці та параметр τ – *незмінні*, то схеми називають *стаціонарними*.

7.1.2. Умови збіжності двошарових ітераційних схем

7.1.2.1. Умови збіжності схеми загального вигляду

Вектори-похибки розв'язку $\{\Delta x\}^{(k)}=\{x\}^{(k)}-\{x\}$ та $\{\Delta x\}^{(k+1)}=\{x\}^{(k+1)}-\{x\}$, з них $\{x\}^{(k)}=\{x\}+\{\Delta x\}^{(k)}$ та $\{x\}^{(k+1)}=\{x\}+\{\Delta x\}^{(k+1)}$. Якщо останні вирази підставити у (7.12), то з урахуванням (7.1) та (7.13) можна легко отримати рівняння

$$\{\Delta x\}^{(k+1)}=[G]^{(k+1)}\{\Delta x\}^{(k)};\quad k=0,1,\ldots. \tag{7.14}$$

З (7.14) утворюється *необхідна* та *достатня* умова збіжності ітераційної схеми (7.7): оскільки модуль вектору-похибки не повинен зростати, то

$$q^{(k+1)}=\|[G]^{(k+1)}\|=\|[I]-\tau^{(k+1)}([Q]^{(k)})^{-1}[A]\|\le 1. \tag{7.15}$$

Чим менші значення $q^{(k+1)}$, тим швидше буде збігатися алгоритм.

Для *стаціонарного* варіанта схеми (7.7)

$$[Q]\frac{\{x\}^{(k+1)}-\{x\}^{(k)}}{\tau}+[A]\{x\}^{(k)}=\{b\};\quad k=0,1,\ldots \tag{7.16}$$

доведено теорему, що вона збігається до точного розв'язку при будь-якому початковому $\{x\}^{(0)}$ тоді і тільки тоді (*необхідна* та *достатня* умови), коли *всі власні значення* матриці переходу $[G]=[I]-\tau\cdot[Q]^{-1}[A]$ за *модулем менше одиниці*. Але користуватися цією умовою незручно, оскільки знаходження *всього* спектру матриці – не менш трудомістка задача, ніж розв'язування СЛАР (7.1).

7.1.2.2. Умови збіжності схеми із симетричною та позитивно визначеною матрицею СЛАР

Для випадку, коли в стаціонарній схемі (7.16) матриця $[A]$ є *симетричною* та *позитивно визначеною*, доведена теорема: алгоритм збігається при будь-якому початковому $\{x\}^{(0)}$, якщо

$$\|[Q]-0.5\tau\cdot[A]\|>0\,,\ \text{або}\ \left([Q]\{x\}^{(k)},\{x\}^{(k)}\right)\geq\frac{\tau}{2}\left([A]\{x\}^{(k)},\{x\}^{(k)}\right). \tag{7.17}$$

При $[Q]=[I]$ перша форма умови (7.17) вироджується в умову $\tau<2/\overline{\lambda}(A)$.

Для аналогічного випадку доведено й іншу теорему: якщо

$$\frac{[Q]^T+[Q]}{2}-\frac{\tau}{2}[A]\geq\frac{(1-q^2)}{2\tau}[Q]^T[A]^{-1}[Q], \tag{7.18}$$

де число $q\in[0,1]$ не залежить від номера ітерації, то стаціонарна схема (7.16) збігається, а для вектора-похибки розв'язку є вірною оцінка

$$\|\{\Delta x\}^{(k)}\|\leq q^k\,\|\{\Delta x\}^{(0)}\|. \tag{7.19}$$

7.1.2.3. Умови збіжності схеми із симетричною та позитивно визначеною матрицею СЛАР і симетричною матрицею розщеплення

Доведено, що у випадку симетричності *обох* матриць $[Q]$ і $[A]$, таких, що

$$\gamma_1[Q]\leq[A]\leq\gamma_2[Q], \tag{7.20}$$

де дійсні числа $\gamma_2>\gamma_1>0$, і при виборі параметра τ як

$$\tau=2/(\gamma_1+\gamma_2), \tag{7.21}$$

стаціонарна ітераційна схема (7.16) збігається, а для вектора-похибки $\{\Delta x\}$ виконується оцінка (7.19), де

$$q=(1-\xi)/(1+\xi);\quad \xi=\gamma_1/\gamma_2. \tag{7.22}$$

Як наслідок цієї теореми

$$\gamma_1\leq\underline{\lambda}(Q^{-1}A);\quad \gamma_2\geq\overline{\lambda}(Q^{-1}A), \tag{7.23}$$

тобто згідно з (7.21) для стаціонарної схеми (7.16) при умові симетричності матриць $[Q]$ і $[A]$ оптимальним значенням параметра τ є значення

$$\widetilde{\tau}=\frac{2}{\underline{\lambda}(Q^{-1}A)+\overline{\lambda}(Q^{-1}A)}=\frac{2}{2-\underline{\lambda}(\overline{G})-\overline{\lambda}(\overline{G})}. \tag{7.24}$$

Тут $\underline{\lambda}()$ та $\overline{\lambda}()$ – відповідно мінімальне та максимальне власні числа вказаних у дужках матриць (про методи їх визначення див. підрозділ 6.3).

Якщо для стаціонарної схеми (7.12), тобто $\{x\}^{(k+1)}=[G]\{x\}^{(k)}+\{\beta\}$, вдається знайти таку матрицю $[W]$, що матриця $[W]\cdot([I]-[G])\cdot[W]^{-1}$ є симетричною та позитивно визначеною ($[W]$ – будь-яка не вироджена матриця *симетрізації*,

наприклад $[W]^T[W]=[Q]$), то така схема називається *такою, що симетрізується*. Тоді всі власні значення матриці $[G]$ є дійсними та $\overline{\lambda}(G)<1$, хоча, можливо, що є від'ємні власні значення, які за модулем перевищують одиницю.

7.1.3. Загальні міркування щодо вибору матриці розщеплення

Оскільки матрицю розщеплення $[Q]$ необхідно швидко обертати, її зазвичай обирають діагональною (кількість дій при обертанні $O(N)$), трикутною ($O(N^2)$ при повному її заповненні) або блочною з трикутними блоками. Схема називається *явною*, коли $[Q]=[I]$, інакше – *неявною*.

Досить часто матрицю розщеплення $[Q]$ збирають з частин матриці $[A]$. Зокрема, коли матрицю $[A]$ представляють у вигляді сукупності її діагональної $[A_D]$, верхньої $[A_U]$ та нижньої $[A_L]$ частин, тобто як

$$[A]=[A_L]+[A_D]+[A_U], \tag{7.25}$$

причому не тільки з окремих компонент, а й з блоків матриці (таких, які можна швидко обернути).

Коли матриця $[Q]=[A_D]$ містить діагональні *блоки* з матриці $[A]$, то метод називається *блочним*, а коли тільки діагональні компоненти, то – *точковим*.

7.1.4. Загальна оцінка кількості ітерацій

Оціночна формула (7.19) дозволяє приблизно розрахувати кількість ітерацій, необхідних для зниження норми вектора похибок наближення $\{\Delta x\}^{(k)}$ на p порядків, відносно норми початкового вектора похибок наближення $\{\Delta x\}^{(0)}$. Позначимо $\varepsilon=10^{-p}$. Тоді з умови $\|\{\Delta x\}^{(k)}\|\le\varepsilon\cdot\|\{\Delta x\}^{(0)}\|$ та з виразу (7.19) випливає, що $q^k\le\varepsilon$. Оскільки $q<1$ та $\varepsilon<1$, то спочатку умову $q^k\le\varepsilon$ записують як $(1/q)^k\ge(1/\varepsilon)$, звідкіля оцінка необхідної кількості ітерацій

$$k\ge\ln(1/\varepsilon)/\ln(1/q). \tag{7.26}$$

Якщо обрано $[Q]=[I]$ (явна схема), то $\underline{\lambda}(Q^{-1}A)=\underline{\lambda}(A)$ і $\overline{\lambda}(Q^{-1}A)=\overline{\lambda}(A)$. При цьому $\gamma_1=\underline{\lambda}(A)$, $\gamma_2=\overline{\lambda}(A)$ і друге відношення з (7.22) $\xi=\underline{\lambda}(A)/\overline{\lambda}(A)=1/\nu_{[A]}$, де $\nu_{[A]}$ – число обумовленості матриці $[A]$ (див. підрозділ 5.2). Коли матриця $[A]$ погано обумовлена, то значення $\nu_{[A]}$ дуже велике, а ξ – мало. При цьому, відповідно до (7.22) і (7.26) $k\ge\ln(1/\varepsilon)/\ln(1/q)\approx\ln(1/\varepsilon)/2\xi=\nu_{[A]}\cdot\ln(1/\varepsilon)/2=O(1/\xi)$, схема збігається дуже повільно. Тому зазвичай намагаються обрати таку матрицю розщеплення $[Q]\ne[I]$, щоб число q було як можна менше одиниці.

Розглянемо декілька окремих випадків, відомих як класичні схеми.

7.1.5. Метод Якобі (J)

Матриця $[Q]=[A_D]$, тобто діагональна стаціонарна. Параметр $\tau=1$, тому з рівняння (7.16):

$$[Q](\{x\}^{(k+1)}-\{x\}^{(k)})+[A]\{x\}^{(k)}=\{b\};\quad k=0,1,\ldots. \tag{7.27}$$

Оскільки в методі Якобі припускається, що обов'язково всі $a_{mm} \neq 0$, то після введення матриці $[\alpha]$ та вектора $\{\beta\}$ з компонентами

$$\alpha_{mn} = \begin{cases} -a_{mn}/a_{mm}; & m \neq n; \\ 0; & m = n; \end{cases} \qquad \beta_m = b_m / a_{mm} \tag{7.28}$$

метод Якобі можна представити у вигляді

$$(x_m)^{(k+1)} = \sum_{m=1}^{N} \alpha_{mn} \cdot (x_m)^{(k)} + \beta_m \quad \text{або} \quad \{x\}^{(k+1)} = [\alpha]\{x\}^{(k)} + \{\beta\}; \quad k = 0,1,\ldots. \tag{7.29}$$

У випадку *симетричної* позитивно визначеної матриці $[A]$, додатково до необхідної та достатньої умови, ще доведені теореми про *достатні* умови збіжності методу Якобі: він збігається при будь-якому початковому векторі $\{x\}^{(0)}$, якщо (див. підрозділ 4.2) норми:

$$\|[\alpha]\|_m = \max_i \sum_j |\alpha_{ij}| < 1 \quad \text{або} \quad \|[\alpha]\|_l = \max_j \sum_i |\alpha_{ij}| < 1 \tag{7.30}$$

та, як наслідок, якщо матриця $[A]$ має так звану *діагональну перевагу*:

$$a_{mm} > \sum_{n=1;\ n\neq m}^{N} |a_{mn}|. \tag{7.31}$$

Примітка 7.1. Якщо умова (7.31) не виконується при тому, що інші умови – виконуються, то вихідну СЛАР, шляхом перестановки невідомих та лінійного комбінування її рядків, завжди можна привести до вигляду, коли умова (7.31) буде виконуватися.

Існує й стаціонарна *екстрапольована* схема методу Якобі (**J-E**). Формула (7.8) приймає вигляд

$$\{x\}^{(k+1)} = \tau \cdot ([\alpha]\{x\}^{(k)} + \{\beta\}) + (1-\tau)\{x\}^{(k)}; \quad k = 0,1,\ldots, \tag{7.32}$$

для якої оптимальне значення $\tau = \tilde{\tau}$ (метод **J-OE**) дається формулою (7.24).

7.1.6. Метод простих ітерацій, метод Річардсона (RF)

Матриця $[Q] = [I]$. Якщо параметр $\tau = 1$, то $[G] = [I] - [A]$ та з (7.16)

$$\{x\}^{(k+1)} = ([I] - [A])\{x\}^{(k)} + \{b\}; \quad k = 0,1,\ldots. \tag{7.33}$$

Це крайній випадок: явна схема. Застосовують замість методу Якобі, коли є хоча б одна компонента $a_{mm} = 0$. Умови збіжності:

- з (7.15) $q = \|([I] - [A])\| \leq 1$ – для загального випадку матриць $[A]$ і $[Q]$;
- з (7.17) $\|[I] - 0.5 \cdot [A]\| > 0$ – для випадку *симетричної* матриці $[A]$. З неї випливає більш зручна умова: $\overline{\lambda}(A) < 2$.

Існує й стаціонарна *екстрапольована* схема методу Річардсона (**RF-E**, 1910 рік). Формула (7.8) приймає вигляд

$$\{x\}^{(k+1)} = ([I] - \tau \cdot [A])\{x\}^{(k)} + \tau\{b\}, \tag{7.34}$$

для якої умови збіжності:

- з (7.15) $q = \|([I] - \tau \cdot [A])\| \leq 1$ – для загального випадку матриць $[A]$ і $[Q]$;

• з (7.17) $\|[I]-0.5\tau\cdot[A]\|>0$ – для випадку *симетричної* матриці $[A]$. З неї випливає більш зручна умова: $\tau<2/\overline{\lambda}(A)$, а оптимальне значення $\tau=\widetilde{\tau}$ (метод **RF-OE**) визначається формулою (7.24): $\widetilde{\tau}=2/(\underline{\lambda}(A)+\overline{\lambda}(A))=2/(2-\underline{\lambda}(G)-\overline{\lambda}(G))$.

У п.7.1.4. було показано, що явні схеми мають малу швидкість збіжності при великих числах обумовленості матриці $[A]$, тобто й навіть метод Річардсона з оптимальним значенням $\tau=\widetilde{\tau}$. Він має лише одну перевагу над методом Якобі: більш універсальний.

7.1.7. Метод верхньої релаксації (SOR)

Симетричну позитивно визначену матрицю $[A]$ представимо у вигляді (7.25), де завдяки симетричності матриці СЛАР $[A_U]=[A_L]^T$. Тоді загальна формула стаціонарного ітераційного процесу (7.16) для методу верхньої релаксації представляється у вигляді:

$$([A_D]+\omega\cdot[A_L])\frac{\{x\}^{(k+1)}-\{x\}^{(k)}}{\omega}+[A]\{x\}^{(k)}=\{b\};\quad k=0,1,\ldots, \tag{7.35}$$

тобто матриця розщеплення $[Q]=([A_D]+\omega[A_L])$ – несиметрична трикутна. Дійсне число ω називають параметром релаксації. Його оптимальне значення рекомендують обчислювати за формулою

$$\omega=2/\left(1+\sqrt{1-(\overline{\lambda}(\overline{G}))^2}\right), \tag{7.36}$$

де матриця $[\overline{G}]=[I]-[A_D]^{-1}[A]$ є матрицею переходу Якобі.

В індексній формі запису метод SOR виглядає значно простіше:

$$(x_m)^{(k+1)}=\omega\cdot\left(-\sum_{n=1}^{m-1}\frac{a_{mn}}{a_{mm}}\cdot(x_n)^{(k+1)}-\sum_{n=m+1}^{N}\frac{a_{mn}}{a_{mm}}\cdot(x_n)^{(k)}+\frac{b_m}{a_{mm}}\right)+(1-\omega)(x_m)^{(k)}, \tag{7.37}$$

де $m=1,2,\ldots,N$; $k=0,1,\ldots$.

Як наслідок умови збіжності (7.17), метод верхньої релаксації збігається при будь-якому початковому $\{x\}^{(0)}$ при $0<\omega<2$ (при $0<\omega<1$ його зазвичай називають методом нижньої релаксації).

7.1.8. Метод Гаусса-Зейделя

Коли матриця $[A]$ є симетричною позитивно визначеною, цей метод можна розглядати як окремий випадок методу верхньої релаксації при $\omega=1$:

$$([A_D]+[A_L])(\{x\}^{(k+1)}-\{x\}^{(k)})+[A]\{x\}^{(k)}=\{b\};\quad k=0,1,\ldots,\ \text{або} \tag{7.38-а}$$

$$([A_D]+[A_L])\{x\}^{(k+1)}+[A_U]\{x\}^{(k)}=\{b\};\quad k=0,1,\ldots. \tag{7.38-б}$$

У цьому випадку метод Гаусса-Зейделя збігається завжди.

Цей метод можна застосовувати й тоді, коли матриця СЛАР несиметрична (формули (7.38) зберігаються). Оскільки $[Q]=[A_D]+[A_L]$, то матриця переходу $[\widetilde{G}]=[I]-([A_D]+[A_L])^{-1}[A]$. Умови збіжності для загального випадку матриць $[A]$ і $[Q]$ наведені в п.п.7.1.2.1 для формули (7.16). У загальному випадку метод не є симетрізуємим. Більш конкретно: метод не можна застосовувати, якщо

матриця переходу $[\widetilde{G}]$ має комплексні власні значення, або якщо її власні вектори не створюють повного базису (див. підрозділ 6.1).

В індексній формі метод Гаусса-Зейделя записується як

$$(x_m)^{(k+1)} = -\sum_{n=1}^{m-1} \frac{a_{mn}}{a_{mm}} \cdot (x_n)^{(k+1)} - \sum_{n=m+1}^{N} \frac{a_{mn}}{a_{mm}} \cdot (x_n)^{(k)} + \frac{b_m}{a_{mm}}; \quad m = 1,2,...,N; \quad k = 0,1,.... \quad (7.39)$$

Отже, тільки що одержане нове значення $(x_n)^{(k+1)}$ в тій же ітерації застосовується для знаходження $(x_m)^{(k+1)}$. Тому цей метод часто вважають модифікацією методу простих ітерацій, з метою його значного прискорення.

Існує й стаціонарна *екстрапольована* схема методу Гаусса-Зейделя. Формула (7.38-а) приймає вигляд

$$([A_D]+[A_L])\frac{\{x\}^{(k+1)} - \{x\}^{(k)}}{\tau} + [A]\{x\}^{(k)} = \{b\}; \quad k = 0,1,..., \quad (7.40)$$

для якої умови збіжності:

- $q = \| ([I] - \tau \cdot ([A_D]+[A_L])^{-1}[A]) \| \le 1$ з (7.15) – для загального випадку матриці $[A]$;
- з (7.17) $\| [I] - 0.5\tau \cdot [A] \| > 0$ – для випадку *симетричної* матриці $[A]$.

Метод більш швидкий, ніж метод Річардсона (простих ітерацій), і достатньо простий у реалізації. Недоліки такі ж, що й у методі простих ітерацій.

7.1.9. Метод симетричної верхньої релаксації (SSOR)

Метод симетричної верхньої релаксації має запис

$$\frac{\omega}{2-\omega}\left(\frac{1}{\omega}[A_D]-[A_L]\right)[A_D]^{-1}\left(\frac{1}{\omega}[A_D]-[A_U]\right)(\{x\}^{(k+1)} - \{x\}^{(k)}) + [A]\{x\}^{(k)} = \{b\}, \quad (7.41)$$

де $k = 0,1,....$ Оскільки $\left(\frac{1}{\omega}[A_D]-[A_L]\right) = \left(\frac{1}{\omega}[A_D]-[A_U]\right)^T$, то схема (7.41) має симетричну матрицю розщеплення: вона є конгруентною до симетричної матриці $[A_D]^{-1}$. Тобто метод є симетризуємим. Як слідство умови збіжності (7.17), метод верхньої релаксації при симетричній матриці $[A]$ збігається при будь-якому початковому $\{x\}^{(0)}$, при $0 < \omega < 2$. Оптимальне значення ω рекомендують обчислювати за формулою

$$\omega = 2/\left(1 + \sqrt{2(1 - \overline{\lambda}(\overline{G}))}\right), \quad (7.42)$$

де матриця $[\overline{G}] = [I] - [A_D]^{-1}[A]$ – матриця переходу Якобі.

Схему часто записують у вигляді сукупності двох релаксаційних схем, які виконуються послідовно в кожній ітерації (в індексній формі):

$$(x_m)^{(k+1/2)} = \omega \cdot \left(-\sum_{n=1}^{m-1} \frac{a_{mn}}{a_{mm}} \cdot (x_n)^{(k+1/2)} - \sum_{n=m+1}^{N} \frac{a_{mn}}{a_{mm}} \cdot (x_n)^{(k)} + \frac{b_m}{a_{mm}} \right) + (1-\omega)(x_m)^{(k)}, \quad (7.43)$$

де $m = 1,2,...,N$; $k = 0,1,...$;

$$(x_m)^{(k+1)} = \omega \cdot \left(-\sum_{n=1}^{m-1} \frac{a_{mn}}{a_{mm}} \cdot (x_n)^{(k+1/2)} - \sum_{n=m+1}^{N} \frac{a_{mn}}{a_{mm}} \cdot (x_n)^{(k+1)} + \frac{b_m}{a_{mm}} \right) + (1-\omega)(x_m)^{(k+1/2)}, \quad (7.44)$$

де $m = N, N-1, ..., 1$; $k = 0, 1, ...$. В них тільки що одержані нові значення $(x_n)^{(k+1/2)}$ і $(x_n)^{(k+1)}$ в тій же ітерації застосовуються для знаходження $(x_m)^{(k+1/2)}$ і $(x_m)^{(k+1)}$.

Оскільки метод SSOR має приблизно у два рази більшу кількість дій, ніж SOR, він не є більш ефективним. Але, оскільки метод є симетризуємим, то можна застосувати екстрапольовану схему:

$$\frac{\omega}{2-\omega}\left(\frac{1}{\omega}[A_D]-[A_L]\right)[A_D]^{-1}\left(\frac{1}{\omega}[A_D]-[A_U]\right)\frac{\{x\}^{(k+1)}-\{x\}^{(k)}}{\tau}+[A]\{x\}^{(k)}=\{b\}, \qquad (7.45)$$

де $k = 0, 1, ...$. Якщо параметр τ можливо знайти з умови, що матриця переходу цієї схеми за модулем менша, ніж 0.25, то швидкість збіжності екстрапольованої схеми SSOR (за кількістю ітерацій) стає приблизно у два рази більша, ніж у методі верхньої релаксації SOR. Тоді екстрапольовану схему SSOR (7.45) відносять до ефективних схем розв'язування СЛАР.

7.1.10. Про блочну реалізацію методів розв'язування СЛАР

Будь-яку СЛАР (7.1) можна представити в блочному вигляді:

$$\begin{bmatrix} A_{11} & A_{12} & ... & A_{1r} \\ A_{21} & A_{22} & ... & A_{2r} \\ \vdots & \vdots & \vdots & \vdots \\ A_{r1} & A_{r2} & ... & A_{rr} \end{bmatrix}\begin{pmatrix} X_1 \\ X_2 \\ \vdots \\ X_r \end{pmatrix}=\begin{pmatrix} B_1 \\ B_2 \\ \vdots \\ B_r \end{pmatrix}, \qquad (7.46)$$

де A_{mn} – квадратні підматриці розміром $N_j \times N_j$, причому $N_1 + N_2 + ... + N_r = N$. Якщо матриця $[A]$ – симетрична та позитивно визначена, то діагональні підматриці A_{mm} теж симетричні та позитивно визначені. Якщо m-й рядок СЛАР (7.46) помножити на вектор невідомих, то можна записати вираз

$$A_{m1}X_1 + A_{m2}X_1 + ... + A_{mr}X_r = \sum_{n=1}^{r} A_{mn}X_n = B_m, \qquad (7.47)$$

який у свою чергу можна переписати відносно m-го діагонального блоку

$$A_{mm}X_m = F_m, \qquad (7.48)$$

де компоненти блочного підвектора

$$F_m = -\sum_{n=1;\ n\neq m}^{r} A_{mn}X_n + B_m. \qquad (7.49)$$

Досить часто блоки A_{mm} мають таку структуру заповнення, яка дозволяє швидко або дуже швидко отримати розв'язок підсистеми (7.48) відносно X_m (зазвичай при відомому F_m) із застосуванням прямого або ітераційного методу. Тоді доцільно замість точкових варіантів методів розв'язування СЛАР (7.1) застосовувати їхні блочні варіанти. Зокрема, метод скінченних різниць генерує матриці з розрідженою структурою та діагональним характером заповнення підматриць. Тоді розв'язок підсистем типа (7.48) можна отримувати із застосуванням методів прогонки (відносяться до прямих методів).

Майже всі схеми розв'язування СЛАР можна представити у блочному вигляді (при $[Q] = [I]$ (метод простих ітерацій, метод Річардсона) – не вдається).

Наприклад, ітераційний метод Якобі (7.29) – як

$$A_{mm}(X_m)^{(k+1)} = -\sum_{n=1;\ n\neq m}^{r} A_{mn}\cdot(X_n)^{(k)} + B_m; \tag{7.50}$$

метод SOR (7.35) – у вигляді

$$A_{mm}(X_m)^{(k+1)} = \omega\cdot\left(-\sum_{n=1}^{m-1} A_{mn}\cdot(X_n)^{(k+1)} - \sum_{n=m+1}^{r} A_{mn}\cdot(X_n)^{(k)} + B_m\right) + (1-\omega)A_{mm}(X_m)^{(k)}, \tag{7.51}$$

а метод Гаусса-Зейделя – як

$$A_{mm}(X_m)^{(k+1)} = -\sum_{n=1}^{m-1} A_{mn}\cdot(X_n)^{(k+1)} - \sum_{n=m+1}^{r} A_{mn}\cdot(X_n)^{(k)} + B_m. \tag{7.52}$$

Тут $m = 1,2,...,r$ – номери блоків-рядків, а $k = 0,1,...$ – номер ітерації.

7.2. Ітераційні методи розв'язування систем лінійних алгебраїчних рівнянь, які використовують варіаційні принципи

Значним недоліком ітераційних методів розв'язування СЛАР, розглянутих у підрозділі 7.1, є використання спектральних характеристик матриці СЛАР: це потребує багато або дуже багато додаткових дій. Методи, які використовують варіаційні принципи, не мають цього недоліку.

Зазвичай використовують канонічну форму двошарової ітераційної схеми (7.7) з незмінними матрицями:

$$[Q]\frac{\{x\}^{(k+1)} - \{x\}^{(k)}}{\tau^{(k+1)}} + [A]\{x\}^{(k)} = \{b\}; \quad k = 0,1,..., \tag{7.53}$$

для якої параметр $\tau^{(k+1)}$ знаходять з умов *мінімуму* норми вектора-похибки наближення $\|\{\Delta x\}^{(k+1)}\|_D = \|\{x\}^{(k+1)} - \{x\}\|_D$, де $\{x\}$ – точний вектор-розв'язок, а деяка матриця $[D]$ є симетричною позитивно визначеною; норма $\|v\|_D = \sqrt{(Dv,v)}$.

У залежності від вибору матриць $[Q]$ та $[D]$ можна отримати різні методи.

7.2.1. Метод мінімальних похибок наближення

Вважається, що матриця СЛАР $[A]$ – симетрична та позитивно визначена, а матриця розщеплення $[Q] = [I]$, тобто одинична. Тоді, якщо застосувати вектор похибок наближення (7.4), то формулу (7.53) можна записати у вигляді рекурентної схеми (див. також рекурентну схему (7.11)):

$$\{x\}^{(k+1)} = \{x\}^{(k)} + \tau^{(k+1)}\{r\}^{(k)}; \quad k = 0,1,.... \tag{7.54}$$

Очевидно, що вектори $\{\Delta x\}^{(k)}$ та $\{r\}^{(k)}$ пов'язані рівнянням

$$[A]\{\Delta x\}^{(k)} = -\{r\}^{(k)}. \tag{7.55}$$

Підставимо $\{r\}^{(k)}$ з (7.55) у праву частину (7.54), віднімемо від правої і лівої частини результату вектор $\{x\}$ і отримаємо, що

$$\{\Delta x\}^{(k+1)} = \{\Delta x\}^{(k)} - \tau^{(k+1)}[A]\{\Delta x\}^{(k)}; \quad k = 0,1,.... \tag{7.56}$$

Помножимо (7.54) зліва на матрицю $[A]$, потім віднімемо від лівої та правої частини вектор $\{b\}$ і отримаємо, що

$$\{r\}^{(k+1)} = \{r\}^{(k)} - \tau^{(k+1)}[A]\{r\}^{(k)}; \quad k = 0,1,.... \tag{7.57}$$

Отже, вектори $\{\Delta x\}^{(k+1)}$ та $\{r\}^{(k+1)}$ задовольняють однаковому рівнянню, тому їх можна використовувати як еквівалентні показники точності розв'язку.

У методі *мінімальних похибок наближення* параметр $\tau^{(k+1)}$ знаходять з умов мінімуму похибки $\|\{r\}^{(k+1)}\|$ при заданій величині похибки $\|\{r\}^{(k)}\|$, тобто при $[D]=[I]$. Для цього зводять (7.57) у квадрат, отримують функціонал (у векторній формі)

$$F=(\vec{r},\vec{r})^{(k+1)}=(\vec{r},\vec{r})^{(k)}-2\tau^{(k+1)}(\vec{r}^{(k)},A\vec{r}^{(k)})+(\tau^{(k+1)})^2(A\vec{r}^{(k)},A\vec{r}^{(k)});\quad k=0,1,\ldots, \quad (7.58)$$

який мінімізують за умовою $\partial F/\partial\tau^{(k+1)}=0$ і отримують вираз для оптимального значення параметра $\tau^{(k+1)}$:

$$\tau^{(k+1)}=(\vec{r}^{(k)},A\vec{r}^{(k)})/(A\vec{r}^{(k)},A\vec{r}^{(k)});\quad k=0,1,\ldots. \quad (7.59)$$

Отже, коли є $\{x\}^{(k)}$, $k=0,1,\ldots$ (у якості початкового $\{x\}^{(0)}$ може бути довільний вектор), то потрібно спочатку обчислити $\{r\}^{(k)}=\{b\}-[A]\{x\}^{(k)}$, потім послідовно застосувати формули (7.59) і (7.54).

Швидкість збіжності цього методу така ж сама, як й екстрапольованого методу Річардсона (див. п. 7.1.6).

7.2.2. Метод мінімальних поправок

Вважається, що матриця СЛАР $[A]$ – симетрична, позитивно визначена, а матриця розщеплення $[Q]\neq[I]$. Тоді, якщо застосувати вектор похибок наближення (7.4) та вектор-поправку (7.10), тобто $\{w\}^{(k)}=[Q]^{-1}\{r\}^{(k)}=\ =[Q]^{-1}(\{b\}-[A]\{x\}^{(k)})$, то формулу (7.53) можна записати як рекурентну схему:

$$\{x\}^{(k+1)}=\{x\}^{(k)}+\tau^{(k+1)}\{w\}^{(k)};\quad k=0,1,\ldots. \quad (7.60)$$

Віднімемо від лівої та правої частини вектор $\{x\}$, використаємо (7.55), тобто $\{\Delta x\}^{(k)}=-[A]^{-1}\{r\}^{(k)}$, помножимо на матрицю $[A]$, отримуємо як проміжний результат, що $\{r\}^{(k+1)}=\{r\}^{(k)}-\tau^{(k+1)}[A]\{w\}^{(k)}$. З (7.10) $\{r\}^{(k)}=[Q]\{w\}^{(k)}$. Позбудемося векторів похибок наближення, помножимо результат зліва на $[Q]^{-1}$ і отримаємо, що

$$\{w\}^{(k+1)}=\{w\}^{(k)}-\tau^{(k+1)}[Q]^{-1}[A]\{w\}^{(k)};\quad k=0,1,\ldots. \quad (7.61)$$

У методі мінімальних поправок параметр $\tau^{(k+1)}$ знаходять з умов *мінімуму* поправки $\|\{w\}^{(k+1)}\|_Q$, при заданій поправці $\|\{w\}^{(k)}\|_Q$, тобто при $[D]=[Q]$. З (7.61) отримують функціонал (у векторній формі)

$$F=(Q\vec{w},\vec{w})^{(k+1)}=(Q\vec{w},\vec{w})^{(k)}-2\tau^{(k+1)}(\vec{w}^{(k)},QQ^{-1}A\vec{w}^{(k)})+(\tau^{(k+1)})^2(QQ^{-1}A\vec{w}^{(k)},Q^{-1}A\vec{w}^{(k)}), \quad (7.62)$$

який мінімізують за умовою $\partial F/\partial\tau^{(k+1)}=0$ і отримують вираз для оптимального значення параметра $\tau^{(k+1)}$:

$$\tau^{(k+1)}=(\vec{w}^{(k)},A\vec{w}^{(k)})/(A\vec{w}^{(k)},Q^{-1}A\vec{w}^{(k)});\quad k=0,1,\ldots. \quad (7.63)$$

Отже, коли є $\{x\}^{(k)}$, $k=0,1,\ldots$, то потрібно спочатку обчислити $\{r\}^{(k)}=\{b\}-[A]\{x\}^{(k)}$, потім $\{w\}^{(k)}=[Q]^{-1}\{r\}^{(k)}$, а потім послідовно застосувати формули (7.63) і (7.60). Швидкість збіжності цього методу така ж сама, як й в екстрапольованому методі Якобі (див. п.7.1.5).

7.2.3. Метод найшвидшого спуску

Вважається, що матриця СЛАР $[A]$ – симетрична, позитивно визначена, а матриця розщеплення $[Q]=[I]$, тобто одинична.

У методі *найшвидшого спуску* параметр $\tau^{(k+1)}$ знаходять з умов *мінімуму* похибки $\|\{\Delta x\}^{(k+1)}\|_A$ при заданій величині похибки $\|\{\Delta x\}^{(k)}\|_A$, тобто при $[D]=[A]$. Для цього з (7.56) отримують функціонал (у векторній формі)

$$F=(A\Delta\vec{x},\Delta\vec{x})^{(k+1)}=(A\Delta\vec{x},\Delta\vec{x})^{(k)}-2\tau^{(k+1)}(A\Delta\vec{x}^{(k)},A\Delta\vec{x}^{(k)})+(\tau^{(k+1)})^2(A^2\Delta\vec{x}^{(k)},A\Delta\vec{x}^{(k)}),\quad(7.64)$$

який мінімізують за умовою $\partial F/\partial\tau^{(k+1)}=0$ і отримують вираз для оптимального значення параметра $\tau^{(k+1)}$:

$$\tau^{(k+1)}=(A\Delta\vec{x}^{(k)},A\Delta\vec{x}^{(k)})/(A^2\Delta\vec{x}^{(k)},A\Delta\vec{x}^{(k)});\quad k=0,1,\ldots.\qquad(7.65)$$

Оскільки $[A]\{\Delta x\}^{(k)}=-\{r\}^{(k)}$, то вираз (7.65) значно спрощується:

$$\tau^{(k+1)}=(\vec{r}^{(k)},\vec{r}^{(k)})/(A\vec{r}^{(k)},\vec{r}^{(k)});\quad k=0,1,\ldots.\qquad(7.66)$$

Отже, коли є $\{x\}^{(k)}$, $k=0,1,\ldots$, то потрібно спочатку обчислити $\{r\}^{(k)}=\{b\}-[A]\{x\}^{(k)}$, потім послідовно застосувати формули (7.66) і (7.54).

Швидкість збіжності цього методу така ж, як й в екстрапольованому методі Річардсона (див. п.7.1.6).

Неявний метод найшвидшого спуску отримують аналогічно, але при $[Q]\neq[I]$ та $[D]=[A]$. Замість (7.56):

$$\{\Delta x\}^{(k+1)}=\{\Delta x\}^{(k)}-\tau^{(k+1)}[Q]^{-1}[A]\{\Delta x\}^{(k)};\quad k=0,1,\ldots.\qquad(7.67)$$

Спочатку одержують $\tau^{(k+1)}=(A\Delta\vec{x}^{(k)},Q^{-1}A\Delta\vec{x}^{(k)})/(AQ^{-1}A\Delta\vec{x}^{(k)},Q^{-1}A\Delta\vec{x}^{(k)})$, а потім, оскільки $[A]\{\Delta x\}^{(k)}=-\{r\}^{(k)}$, то $[Q]^{-1}[A]\{\Delta x\}^{(k)}=-[Q]^{-1}\{r\}^{(k)}=-\{w\}^{(k)}$ і вираз для $\tau^{(k+1)}$ значно спрощується:

$$\tau^{(k+1)}=(\vec{r}^{(k)},\vec{w}^{(k)})/(A\vec{w}^{(k)},\vec{w}^{(k)});\quad k=0,1,\ldots.\qquad(7.68)$$

Отже, коли є $\{x\}^{(k)}$, $k=0,1,\ldots$, то потрібно спочатку обчислити $\{r\}^{(k)}=$ $=\{b\}-[A]\{x\}^{(k)}$, потім $\{w\}^{(k)}=[Q]^{-1}\{r\}^{(k)}$, а потім послідовно застосувати формули (7.68) і (7.60).

Швидкість збіжності цього методу така ж, як й в екстрапольованому методі Якобі (див. п.7.1.5).

7.2.4. Метод спряжених градієнтів

Метод спряжених градієнтів (Method of conjugate gradients, абревіатура **CG**. Автори: Хестенс (M.R. Hestenes) і Штіфель (E.L. Stiefel), 1952 р.) – велике сімейство алгоритмів. Будемо вважати, що матриці $[Q]$ та $[A]$ – симетричні позитивно визначені (є й варіанти для несиметричної матриці $[A]$). Метод використовує вагову *тришарову* схему, яка в канонічній формі має вигляд (у класичному варіанті приймають $[Q]=[I]$):

$$[Q]\frac{(\{x\}^{(k+1)}-\{x\}^{(k)})+(1-\alpha^{(k+1)})(\{x\}^{(k)}-\{x\}^{(k-1)})}{\tau^{(k+1)}\alpha^{(k+1)}}+[A]\{x\}^{(k)}=\{b\};\quad k=0,1,\ldots,\qquad(7.69)$$

причому при $k=0$ (перший крок) призначається $\alpha^{(1)}=1$, тобто схема має вигляд

$$[Q]\frac{\{x\}^{(1)}-\{x\}^{(0)}}{\tau^{(1)}}+[A]\{x\}^{(0)}=\{b\}. \qquad (7.70)$$

Канонічну форму (7.69) можна представити у вигляді рекурентної схеми

$$\{x\}^{(k+1)}=\alpha^{(k+1)}[\{x\}^{(k)}+\tau^{(k+1)}\{w\}^{(k)}]+(1-\alpha^{(k+1)})\{x\}^{(k-1)};\quad k=0,1,\ldots, \qquad (7.71)$$

де як і раніше $\{w\}^{(k)}=[Q]^{-1}\{r\}^{(k)}$ та $\{r\}^{(k)}=\{b\}-[A]\{x\}^{(k)}$.

Відносно вектора-похибки $\{\Delta x\}^{(k+1)}$ формула буде мати вигляд:

$$\{\Delta x\}^{(k+1)}=\alpha^{(k+1)}([I]-\tau^{(k+1)}[Q]^{-1}[A])\{\Delta x\}^{(k)}+(1-\alpha^{(k+1)})\{\Delta x\}^{(k-1)};\quad k=0,1,\ldots, \qquad (7.72)$$

де при $k=0$ (перший крок) надається значення $\alpha^{(1)}=1$, тобто схема має вигляд

$$\{\Delta x\}^{(1)}=([I]-\tau^{(1)}[Q]^{-1}[A])\{\Delta x\}^{(0)}. \qquad (7.73)$$

Введемо допоміжний вектор $\{v\}^{(k)}=[A]^{1/2}\{\Delta x\}^{(k)}$, який, відповідно до (7.72) та (7.73), буде задовольняти рівнянням

$$\{v\}^{(k+1)}=\alpha^{(k+1)}([I]-\tau^{(k+1)}[C])\{v\}^{(k)}+(1-\alpha^{(k+1)})\{v\}^{(k-1)};\quad k=1,2,\ldots, \qquad (7.74)$$

$$\{v\}^{(1)}=([I]-\tau^{(1)}[C])\{v\}^{(0)}, \qquad (7.75)$$

де матриця $[C]=[A]^{1/2}[Q]^{-1}[A]^{1/2}$ – симетрична та позитивно визначена, як і матриці $[Q]$ та $[A]$.

Якщо з рівняння (7.74) послідовно виключати вектори $\{v\}^{(1)},\{v\}^{(2)},\ldots,\{v\}^{(k-1)}$, можна у підсумку отримати, що

$$\{v\}^{(k)}=P([C])\{v\}^{(0)}, \qquad (7.76)$$

де $P([C])$ – матричний многочлен степені k, причому $P([0])=[I]$. Далі ставиться завдання: обрати параметри $\alpha^{(k+1)}$ і $\tau^{(k+1)}$ таким чином, щоб отримати мінімальне значення норми $\|\{v\}^{(k)}\|=\|\{\Delta x\}^{(k)}\|_A$ в *кожній* ітерації.

Для параметра $\tau^{(1)}$, як й у методі найшвидшого спуску, можна отримати

$$\tau^{(1)}=(C\vec{v}^{(0)},\vec{v}^{(0)})/(C\vec{v}^{(0)},C\vec{v}^{(0)}), \qquad (7.77)$$

причому буде виконуватися умова $(C\vec{v}^{(1)},\vec{v}^{(0)})=0$, тобто $(\vec{v}^{(1)},\vec{v}^{(0)})_C=0$, що вказує на *ортогональність* векторів $\{v\}^{(1)}$ та $\{v\}^{(0)}$.

Для k-й ітерації матричний многочлен $P([C])$ записують у вигляді

$$P([C])=[I]+\sum_{j=1}^{k}s_j^{(k)}[C]^j, \qquad (7.78)$$

де $s_j^{(k)}$ – числові коефіцієнти, залежні від $\alpha^{(j)}$ і $\tau^{(j)}$. Тоді (7.76) отримає вигляд

$$\{v\}^{(k)}=\{v\}^{(0)}+\sum_{j=1}^{k}s_j^{(k)}[C]^j\{v\}^{(0)};\quad k=1,2,\ldots. \qquad (7.79)$$

З (7.79) можна отримати функціонал (у векторній формі)

$$F=(\vec{v}^{(k)},\vec{v}^{(k)})=(\vec{v}^{(0)},\vec{v}^{(0)})+\sum_{i,j=1}^{k}s_i^{(k)}s_j^{(k)}(C^i\vec{v}^{(0)},C^j\vec{v}^{(0)})+2\sum_{j=1}^{k}s_j^{(k)}(\vec{v}^{(0)},C^j\vec{v}^{(0)}), \qquad (7.80)$$

який є многочленом другої степені відносно змінних $s_j^{(k)}$. З умови $\partial F/\partial s_j^{(k)}=0$, $j=1,2,\ldots,k$ отримують систему рівнянь

$$\sum_{i=1}^{k}s_i^{(k)}(C^i\vec{v}^{(0)},C^j\vec{v}^{(0)})+(\vec{v}^{(0)},C^j\vec{v}^{(0)})=0;\quad j=1,2,\ldots,k \qquad (7.81)$$

відносно оптимальних значень $s_i^{(k)}$. Їх необхідно пов'язати з параметрами $\alpha^{(k+1)}$ і $\tau^{(k+1)}$, які теж будуть оптимальними (будуть мінімізувати (7.80)).

Систему рівнянь (7.81) можна записати у вигляді

$$(\vec{v}^{(k)}, C^j\vec{v}^{(0)}) = 0, \quad k = 1,2,\ldots; \quad j = 1,2,\ldots,k. \tag{7.82}$$

Доведено лему, що умови (7.82) еквівалентні умовам

$$(\vec{v}^{(k)}, C\vec{v}^{(j)}) = 0, \quad k = 1,2,\ldots; \quad j = 0,1,\ldots,k-1. \tag{7.83}$$

Ці умови означають, що потрібно знайти ортогональну систему векторів $\{v\}^{(k)}$, кількість яких не буде перевишувати N (кількості рівнянь у СЛАР). Тобто, починаючи з деякої ітерації, вектори похибки $\{v\}^{(k)}$ будуть (теоретично) нульовими, розв'язок СЛАР буде отримано не більш ніж за N ітерацій (практично це часто не відбувається, оскільки у ЕОМ є "обрізання" дійсних чисел).

З (7.74) витікає, що

$$(\vec{v}^{(k+1)}, C^j\vec{v}^{(j)}) = -\alpha^{(k+1)}\tau^{(k+1)}(C\vec{v}^{(k)}, C^j\vec{v}^{(j)}), \quad k = 1,2,\ldots; \quad j = 0,1,\ldots,k-2. \tag{7.84}$$

$$(C\vec{v}^{(k)}, C\vec{v}^{(j)}) = 0, \quad k = 1,2,\ldots; \quad j = 0,1,\ldots,k-2. \tag{7.85}$$

Залишають лише дві умови:

$$(\vec{v}^{(k+1)}, C\vec{v}^{(k)}) = 0; \quad k = 0,1,\ldots; \quad (\vec{v}^{(k+1)}, C\vec{v}^{(k-1)}) = 0; \quad k = 1,2,\ldots. \tag{7.86}$$

Якщо в них підставити (7.74), то з урахуванням (7.83) і (7.85) отримуємо інший вигляд цих умов:

$$\alpha^{(k+1)}(C\vec{v}^{(k)}, \vec{v}^{(k)}) - \alpha^{(k+1)}\tau^{(k+1)}(C\vec{v}^{(k)}, C\vec{v}^{(k)}) + (1-\alpha^{(k+1)})(C\vec{v}^{(k)}, \vec{v}^{(k-1)}) = 0; \quad k = 0,1,\ldots. \tag{7.87}$$

$$-\alpha^{(k+1)}\tau^{(k+1)}(C\vec{v}^{(k)}, C\vec{v}^{(k-1)}) + (1-\alpha^{(k+1)})(C\vec{v}^{(k-1)}, \vec{v}^{(k-1)}) = 0; \quad k = 1,2,\ldots. \tag{7.88}$$

У рівнянні (7.87) остання складова дорівнює нулю відповідно до (7.83). Тому з (7.87) утворюється вираз для оптимального значення параметра $\tau^{(k+1)}$:

$$\tau^{(k+1)} = (C\vec{v}^{(k)}, \vec{v}^{(k)})/(C\vec{v}^{(k)}, C\vec{v}^{(k)}); \quad k = 0,1,\ldots. \tag{7.89}$$

З (7.74) і (7.83) можна одержати, що

$$(C\vec{v}^{(k)}, C\vec{v}^{(k-1)}) = -(C\vec{v}^{(k)}, \vec{v}^{(k)})/\alpha^{(k)}\tau^{(k)}; \quad k = 1,2,\ldots. \tag{7.90}$$

Тоді з (7.88), після виключення $(C\vec{v}^{(k)}, C\vec{v}^{(k-1)})$, оптимальне значення параметра $\alpha^{(k+1)}$:

$$\alpha^{(k+1)} = \left[1 - \frac{\tau^{(k+1)}}{\alpha^{(k)}\tau^{(k)}} \cdot \frac{(C\vec{v}^{(k)}, \vec{v}^{(k)})}{(C\vec{v}^{(k-1)}, \vec{v}^{(k-1)})}\right]^{-1}; \quad k = 1,2,\ldots; \quad \alpha^{(1)} = 1. \tag{7.91}$$

Залишається виключити з рівнянь (7.89) і (7.91) штучно введені матрицю $[C] = [A]^{1/2}[Q]^{-1}[A]^{1/2}$ і допоміжний вектор $\{v\}^{(k)} = [A]^{1/2}\{\Delta x\}^{(k)}$.

Оскільки $\{w\}^{(k)} = [Q]^{-1}\{r\}^{(k)}$ та $\{r\}^{(k)} = \{b\} - [A]\{x\}^{(k)}$, то $(C\vec{v}^{(k)}, C\vec{v}^{(k)}) = (A\vec{w}^{(k)}, \vec{w}^{(k)})$ і

$$\tau^{(k+1)} = (\vec{w}^{(k)}, \vec{r}^{(k)})/(A\vec{w}^{(k)}, \vec{w}^{(k)}); \quad k = 0,1,\ldots. \tag{7.92}$$

$$\alpha^{(k+1)} = \left[1 - \frac{\tau^{(k+1)}}{\alpha^{(k)}\tau^{(k)}} \cdot \frac{(\vec{w}^{(k)}, \vec{r}^{(k)})}{(A\vec{w}^{(k-1)}, \vec{w}^{(k-1)})}\right]^{-1}; \quad k = 1,2,\ldots; \quad \alpha^{(1)} = 1. \tag{7.93}$$

Дуже часто метод спряжених градієнтів записують дещо інакше:

$$\{x\}^{(k+1)} = \{x\}^{(k)} + \lambda^{(k+1)}\{p\}^{(k)}; \quad k = 0,\ 1,\ldots, \tag{7.94}$$

де

$$\{p\}^{(k)} = \{w\}^{(k)} + \beta^{(k)}\{p\}^{(k-1)}; \quad k = 0,1,\ldots; \tag{7.95}$$

$$\lambda^{(k+1)} = \frac{\gamma^{(k)}}{(\vec{p}^{(k)},\ A\vec{p}^{(k)})}; \quad k = 0,\ 1,\ldots; \tag{7.96}$$

$$\beta^{(k)} = \gamma^{(k)} / \gamma^{(k-1)}; \quad k = 1,2,\ldots; \quad \beta^{(0)} = 0; \tag{7.97}$$

$$\gamma^{(k)} = (\{w\}^{(k)},\ \{w\}^{(k)}); \quad k = 0,1,\ldots; \tag{7.98}$$

$$\{w\}^{(k)} = \{w\}^{(k-1)} - \lambda^{(k)}[A]\{p\}^{(k-1)}; \quad k = 1,2,\ldots.. \tag{7.99}$$

Але, якщо з виразів $\{x\}^{(k+1)} = \{x\}^{(k)} + \lambda^{(k)}\{p\}^{(k)}$ та $\{x\}^{(k)} = \{x\}^{(k-1)} + \lambda^{(k-1)}\{p\}^{(k-1)}$ (див. формулу (7.94)) за допомогою формули (7.95) виключити "вектори напрямків" $\{p\}^{(k)}$ та $\{p\}^{(k-1)}$, то після введення позначень

$$\alpha^{(k+1)} = 1 + \lambda^{(k+1)}\beta^{(k)} / \lambda^{(k)}; \quad \tau^{(k+1)} = \lambda^{(k+1)} / \alpha^{(k+1)} \tag{7.100}$$

отримуємо рекурентну схему (7.71), тобто (7.94) ... (7.99) – лише інша (зручна) форма представлення цього методу. "Вектори напрямків" $\{p\}^{(k)}$ та $\{p\}^{(k-1)}$ є A-спряженими, тобто $(\vec{p}^{(k)},\ A\vec{p}^{(k-1)}) = 0$.

Формула (7.99) для обчислення вектора $\{w\}^{(k)}$ є теоретично точною і її зручно використовувати, оскільки допоміжний вектор $[A]\{p\}^{(k-1)} = \{g\}^{(k-1)}$ вже обчислювався в попередній ітерації для одержання значення $\lambda^{(k+1)}$ (див. формулу (7.96)). Але на практиці при такому обчисленні $\{w\}^{(k)}$ накоплюються похибки, тому рекомендують через деяку кількість ітерацій обчислювати його за "прямою" формулою: $\{w\}^{(k)} = [Q]^{-1}\{r\}^{(k)} = [Q]^{-1}(\{b\} - [A]\{x\}^{(k)})$.

Якщо у якості матриці розщеплення $[Q]$ використовувати одиничну матрицю $[I]$, то $\{w\}^{(k)} = \{r\}^{(k)}$.

Для вектора-похибки буде виконуватися оцінка

$$\| \{\Delta x\}^{(k)} \| \le q^{(\bar{k})} \| \{\Delta x\}^{(0)} \|, \tag{7.101}$$

де позначено

$$q^{(\bar{k})} = 2\rho_1^{\bar{k}} / (1 + \rho_1^{2\bar{k}}); \quad \rho_1 = (1 - \sqrt{\xi}) / (1 + \sqrt{\xi}); \tag{7.102}$$

$\bar{k}$ – кількість використаних ітерацій; $\xi = \gamma_1 / \gamma_2$, а γ_1 і γ_2 відповідають (7.20).

Увага: ця швидкість збіжності значно перевищує відповідні показники всіх раніше розглянутих методів.

7.3. Метод спряжених напрямків

Якщо побудувати систему із J взаємно ортогональних A-спряжених векторів $\{p\}^{(n)}$; $n = 0,1,\ldots,N-1$, то вони будуть лінійно незалежними, коли м атриця СЛАР є симетричною позитивно визначеною. Тому на них можна будувати вектор-розв'язок

$$\{x\} = \{x\}^{(0)} + \lambda^{(1)}\{p\}^{(1)} + \ldots + \lambda^{(N-1)}\{p\}^{(N-1)}, \tag{7.103}$$

де $\{x\}^{(0)}$ – деяке призначене початкове наближення. Якщо вираз (7.103) помножити зліва на матрицю $[A]$, а потім обчислювати скалярні добутки з векторами $\{p\}^{(n)}$, то можна отримати вирази для обчислювання коефіцієнтів у (7.103):

$$\lambda^{(n)} = \frac{(\vec{p}^{(n)},\, \vec{b} - A\vec{x}^{(0)})}{(\vec{p}^{(n)},\, A\vec{p}^{(n)})} = \frac{(\vec{p}^{(n)},\, \vec{r}^{(0)})}{(\vec{p}^{(n)},\, A\vec{p}^{(n)})}; \quad n = 0,\, 1,..., N-1. \tag{7.104}$$

Доведено, що можна застосовувати й інший вираз:

$$\lambda^{(n)} = \frac{(\vec{p}^{(n)},\, \vec{r}^{(n)})}{(\vec{p}^{(n)},\, A\vec{p}^{(n)})}; \quad n = 0,\, 1,..., N-1. \tag{7.105}$$

Відповідно до цих формул у методі спряжених напрямків (МСН) вектор-розв'язок $\{x\}$ будується крок за кроком ($k = 0,\, 1,...$ – аналог ітерацій) послідовністю дій:

$$\{r\}^{(k)} = \{b\} - [A]\{x\}^{(k)}; \tag{7.106}$$

$$\lambda^{(k)} = \frac{(\vec{p}^{(k)},\, \vec{r}^{(k)})}{(\vec{p}^{(k)},\, A\vec{p}^{(k)})}; \tag{7.107}$$

$$\{x\}^{(k+1)} = \{x\}^{(k)} + \lambda^{(k)}\{p\}^{(k)}. \tag{7.108}$$

Як обирати вектори $\{p\}^{(k)}$? Єдині вимоги: вектори $\{p\}^{(k)}$ лінійно-незалежні, ортогональні та A-спряжені. Тому метод спряжених градієнтів (див. п.7.2.4) є окремим випадком МСН, коли вектори $\{p\}^{(k)}$ обчислюються як A-спряжені поточні вектори похибок наближення $\{r\}^{(k)}$. Відомі й інші варіанти. Зокрема, за допомогою процедури Грама-Шмідта (див. п.5.3.8) знаходяться N лінійно-незалежних ортогональних векторів $\{y\}^{(n)}$, які потім A-спрягаються. Інший варіант: вектори $\{y\}^{(n)}$ обираються у вигляді одиничних векторів базису, тобто $\{y\}^{(0)} = \{1, 0, ..., 0\}^T$; $\{y\}^{(1)} = \{0, 1, ..., 0\}^T$, ... Такий варіант МСН буде аналогом методу виключення Гаусса (див. п.5.3.3).

7.4. Прискорення ітераційних схем розв'язування СЛАР

Як виявилося, швидкість збіжності багатьох ітераційних схем розв'язування СЛАР можна значно прискорювати.

7.4.1. Поліноміальне прискорення

З другого виразу (7.16) при $\tau = 1$ для *стаціонарного* процесу випливає, що в ітераціях вектори-похибки $\{\Delta x\}$ створюють послідовність векторів $\{\Delta x\}^{(k)}$, і в k-й ітерації

$$\{\Delta x\}^{(k)} = [G]^k \{\Delta x\}^{(0)}; \quad k = 0, 1, ... \tag{7.109}$$

Для прискорення збіжності ітераційного процесу можна застосувати іншу послідовність:

$$\{\Delta x\}^{(k)} = \left(\sum_{i=0}^{k} c_i^{(k)} \cdot [G]^{(k)} \right) \{\Delta x\}^{(0)} = P^{(k)}(G)\{\Delta x\}^{(0)} \tag{7.110}$$

при умовах:

$$\sum_{i=0}^{k} c_i^{(k)} \equiv 1; \quad P^{(k)}(I) = [I]; \quad k = 0,1,..., \tag{7.111}$$

яка забезпечує $\{x\}^{(k)} = \{x\}$, коли було обрано $\{x\}^{(0)} = \{x\}$ (тут $\{x\}$ – точний вектор-розв'язок). Матричний многочлен $P^{(k)}(G)$ має вигляд:

$$P^{(k)}(G) = c_0^{(k)}[I] + c_1^{(k)}[G] + c_2^{(k)}[G]^2 + ... + c_k^{(k)}[G]^k. \tag{7.112}$$

Але пряме застосування (7.112) потребує забагато дій, оскільки $[G]^k$ є результатом перемноження матриці $[G]$ k разів. Виявилося, що матричний многочлен $P^{(k)}(G)$ має безліч еквівалентних замін, які задовольняють умовам (7.111). Зокрема, доведено теорему, що це може бути *тришарова* ітераційна схема:

$$\{x\}^{(k+1)} = \rho^{(k+1)}(\gamma^{(k+1)}([G]\{x\}^{(k)} + \{\beta\}) + (1-\gamma^{(k+1)})\{x\}^{(k)}) + (1-\rho^{(k+1)})\{x\}^{(k-1)}; \; k > 0, \tag{7.113}$$

де $\rho^{(k+1)}, \gamma^{(k+1)}$ – будь-які дійсні числа, а в першій ітерації використовується формула

$$\{x\}^{(1)} = \gamma^{(1)}([G]\{x\}^{(0)} + \{\beta\}) + (1-\gamma^{(1)})\{x\}^{(0)}. \tag{7.114}$$

Отже, немає потреби в ітераційних схемах з більшою кількістю шарів, ніж з трьома шарами.

7.4.2. Чебишевське прискорення

Застосовують до стаціонарних симетрізуємих ітераційних схем у вигляді (7.16) при $\tau = 1$. Для них $\overline{\lambda}(G) < 1$.

У тришаровій ітераційній схемі (7.113) поліноміального прискорення формально $\rho^{(k+1)}, \gamma^{(k+1)}$ – будь-які дійсні числа. Але їх вдалий вибір може призвести до значного додаткового прискорення схеми. Зокрема, схемою *з оптимальним чебишевським прискоренням* називається схема (7.113) з параметрами

$$\gamma^{(k+1)} = \widetilde{\gamma} = 2/(2 - \overline{\lambda}(G) - \underline{\lambda}(G)); \tag{7.115}$$

$$\rho^{(k+1)} = \widetilde{\rho}^{(k+1)} = 2\widetilde{\omega} \cdot T^{(n)}(\widetilde{\omega}) / T^{(n+1)}(\widetilde{\omega}), \tag{7.116}$$

де $\widetilde{\omega} = (2 - \overline{\lambda}(G) - \underline{\lambda}(G))/(\overline{\lambda}(G) - \underline{\lambda}(G))$, а $T^{(n)}(\widetilde{\omega})$ – многочлен Чебишева

$$T^{(n)}(\omega) = \cos(n \cdot \arccos\omega) \quad \text{при} \quad -1 \le \omega \le 1. \tag{7.117}$$

Замість формули (7.116) зазвичай застосовують інше представлення цього співвідношення: у вигляді рекурентної формули

$$\rho^{(1)} = 1; \quad \widetilde{\rho}^{(2)} = 1/(1 - 0.5\widetilde{\sigma}^2); \quad \widetilde{\rho}^{(k+1)} = 1/(1 - 0.25\widetilde{\sigma}^2\widetilde{\rho}^{(k)}); \quad k = 2,3,..., \tag{7.118}$$

де позначено $\widetilde{\sigma} = 1/\widetilde{\omega} = (\overline{\lambda}(G) - \underline{\lambda}(G))/(2 - \overline{\lambda}(G) - \underline{\lambda}(G))$.

Але частіше чебишевське прискорення застосовують до двошарових ітераційних схем.

Екстрапольовану схему методу Річардсона (7.34) з симетричною позитивно визначеною матрицею $[A]$ запишемо зі змінним параметром $\tau^{(k+1)}$:

$$\frac{\{x\}^{(k+1)} - \{x\}^{(k)}}{\tau^{(k+1)}} + [A]\{x\}^{(k)} = \{b\}; \quad k = 0,1,.... \tag{7.119}$$

Доведено теорему, що ця схема за вказану кількість ітерацій $k = \overline{k}$ знайде вектор-розв'язок з найменшою похибкою, якщо в ітераціях призначати:

$$\tau^{(k)} = \tau_0/(1 + \rho_0 t^{(k)}); \quad k = 1,2,..., \overline{k}, \tag{7.120}$$

де параметри

$$\begin{cases} \tau_0 = 2/(\underline{\lambda}(A) + \overline{\lambda}(A)); \quad \rho_0 = (1-\zeta)/(1+\zeta); \\ \zeta = \underline{\lambda}(A)/\overline{\lambda}(A); \qquad t^{(k)} = \cos((2k-1)\pi/2k). \end{cases} \tag{7.121}$$

Така схема називається явною ітераційною схемою з чебишевським набором параметрів.

Для вектор-похибки буде виконуватися оцінка (7.101), тобто швидкість збіжності цієї схеми така ж, як і у методі спряжених градієнтів.

Оціночна формула (7.101) дозволяє приблизно розрахувати кількість ітерацій, необхідних для зниження норми початкового вектору похибок наближення $\{\Delta x\}^{(0)}$ на p порядків (див. п. 7.1.4). Позначимо $\varepsilon = 10^{-p}$. Тоді з умови $\| \{\Delta x\}^{(k)} \| \le \varepsilon \cdot \| \{\Delta x\}^{(0)} \|$ та з виразу (7.101) випливає, що $q^{(k)} \le \varepsilon$, де $q^{(k)}$ відповідає першої формулі (7.102), тобто $2\rho_1^k/(1+\rho_1^{2k}) \le \varepsilon$. Якщо позначити $z = 1/\rho_1^k$, то $z > (1+\sqrt{1-\varepsilon^2})/\varepsilon$. Ця нерівність буде виконана, якщо призначити $z \ge 2/\varepsilon$. Отже, оцінка необхідної кількості ітерацій

$$k \ge \ln(2/\varepsilon)/\ln(1/\rho_1). \tag{7.122}$$

При погано обумовленій матриці $[A]$ відношення $\xi = \underline{\lambda}(A)/\overline{\lambda}(A)$ буде дуже незначним, $\ln(1/\rho_1) = \ln\left((1+\sqrt{\xi})/(1-\sqrt{\xi})\right) \approx 2\sqrt{\xi}$, тобто у цьому випадку

$$k \ge \ln(2/\varepsilon)/2\sqrt{\xi} = O(1/\sqrt{\xi}). \tag{7.123}$$

У п. 7.1.4. було показано, що звичайна *явна* ітераційна схема (7.59) при погано обумовленій матриці $[A]$ потребує кількість ітерацій $k \approx O(1/\xi)$, тобто значно більше, ніж схема (7.119) з чебишевським прискоренням.

Для схеми Якобі (неявної), якщо представити її у вигляді

$$[A_D]\frac{\{x\}^{(k+1)} - \{x\}^{(k)}}{\tau^{(k+1)}} + [A]\{x\}^{(k)} = \{b\}; \quad k = 0,1,..., \tag{7.124}$$

практично всі формули та оцінки (7.101), (7.102), (7.120) ... (7.123) зберігаються, за виключенням того, що $\tau_0 = 2/(\underline{\lambda}(D_D^{-1}A) + \overline{\lambda}(D_D^{-1}A))$ та $\xi = \underline{\lambda}(D_D^{-1}A)/\overline{\lambda}(D_D^{-1}A)$. Оскільки зазвичай число ξ є більшим, ніж у явному методі, то схема (7.124) збігається швидше. Така схема називається неявною ітераційною схемою з чебишевським набором параметрів.

Коли точні границі спектра $\underline{\lambda}()$ і $\overline{\lambda}()$ невідомі, то замість них використовують деякі наближені значення, зокрема, числа γ_1 і γ_2, що відповідають умовам (7.23). Оцінки швидкості збіжності (7.122) та (7.123) зберігаються, хоча оптимальність схеми не гарантується. Як виявилося, швидкість збіжності схеми досить сильно (навіть на десятки відсотків) залежить від похибки визначення $\overline{\lambda}()$, а от від $\underline{\lambda}()$ – незначно (на відсотки). В книзі [77] розглядаються деякі спеціальні алгоритми та оцінки для уточнення чисел γ_1 і γ_2, з метою збереження високої швидкості збіжності ітераційних схем у цьому випадку.

Примітка 7.2. Застосування схем з чебишевським прискоренням може привести до аварійного завершення роботи алгоритму в зв'язку з переповненням (досягненням машинної нескінченності). Це тому, що чебишевське прискорення

не гарантує монотонності у зменшенні похибки розв'язку від ітерації до ітерації. Було доведено, що чебишевські параметри можна застосовувати у будь-якому порядку, не порушуючи оптимальності схеми, після чого було розроблено алгоритми такого порядку обирання чебишевських параметрів, при якому аварійна ситуація не виникає (див., наприклад, книги [66, 77]).

7.4.3. Прискорення за методом спряжених градієнтів

Метод спряжених градієнтів можна розглядати й як один з методів прискорення двошарових ітераційних схем.

Розглянемо один з таких дуже ефективних алгоритмів: алгоритм симетричної релаксації з прискоренням за методом спряжених градієнтів, викладений у книзі [79]. Він потребує симетричну позитивно визначену матрицю СЛАР.

Спочатку СЛАР (7.1) нормують згідно з алгоритмом підрозділу 7.5. Отриману нормовану СЛАР $[\widetilde{A}]\{\widetilde{x}\} = \{\widetilde{b}\}$ (формула (7.145)) можна представити у еквівалентному вигляді:

$$[\overline{A}]\{\overline{x}\} = \{\overline{b}\}, \tag{7.125}$$

де позначені матриця та вектори

$$[\overline{A}] = \omega \cdot \left(\{[I] + [\widetilde{L}_\omega]^T\right)^{-1} [\widetilde{A}] \left(\{[I] + [\widetilde{L}_\omega]\right)^{-1}; \tag{7.126}$$

$$\{\overline{x}\} = \left(\{[I] + [\widetilde{L}_\omega]\right)\{\widetilde{x}\}; \tag{7.127}$$

$$\{\overline{b}\} = \omega \cdot \left(\{[I] + [\widetilde{L}_\omega]^T\right)^{-1} \{\widetilde{b}\}, \tag{7.128}$$

а точні верхня та нижня трикутні матриці

$$[\widetilde{L}_\omega] = \omega \cdot [\widetilde{L}]; \quad [\widetilde{L}_\omega]^T = \omega \cdot [\widetilde{L}]^T, \tag{7.129}$$

де ω – параметр релаксації, а $[\widetilde{L}] + [\widetilde{L}]^T + [I] = [\widetilde{A}]$.

Нехай $\{\widetilde{x}\}^{(0)}$ – довільне початкове наближення, а $\{\overline{r}\}^{(0)}$ – початковий вектор похибки наближення, що обраховується за формулою

$$\{\overline{r}\}^{(0)} = \omega \cdot \left(\{\widetilde{x}\}^{(0)} - \{\widetilde{b}\}\right) + [\widetilde{L}_\omega]\{x\}^{(0)} + [\widetilde{L}_\omega]^T \left(\{\widetilde{x}\}^{(0)} - \{\overline{r}\}^{(0)}\right). \tag{7.130}$$

У даному випадку (та в аналогічних випадках нижче) немає протиріч: у правій частині формули застосовуються тільки ті компоненти вектора $\{\overline{r}\}^{(0)}$, що вже обчислені, оскільки $[\widetilde{L}_\omega]^T$ – точна нижня трикутна матриця, тобто з нулями на діагоналі.

Ще призначається вектор $\{\overline{g}\}^{(0)} = \{\overline{r}\}^{(0)}$ і обчислюється

$$\gamma^{(0)} = (\{\overline{r}\}^{(0)}, \{\overline{r}\}^{(0)}). \tag{7.131}$$

Потім у ітераціях $(k = 0,1,...)$ обчислюються:

$$\{p\} = \{\overline{g}\}^{(k)} - [\widetilde{L}_\omega]\{p\}; \tag{7.132}$$

$$\{q\} = \{\overline{g}\}^{(k)} + (\omega - 1) \cdot \{p\} + [\widetilde{L}_\omega]^T \left(\{p\} - \{q\}\right); \tag{7.133}$$

$$\lambda^{(k)} = \frac{\gamma^{(k)}}{(\{q\}, \{\overline{g}\}^{(k)})}; \tag{7.134}$$

$$\{\widetilde{x}\}^{(k+1)} = \{\widetilde{x}\}^{(k)} - \lambda^{(k)}\{p\}; \tag{7.135}$$

$$\{\overline{r}\}^{(k+1)} = \{\overline{r}\}^{(k)} - \lambda^{(k)}\{q\}; \tag{7.136}$$

$$\gamma^{(k+1)} = (\{\bar{r}\}^{(k+1)}, \{\bar{r}\}^{(k+1)}); \tag{7.137}$$

$$\alpha^{(k+1)} = \gamma^{(k+1)} / \gamma^{(k)}; \tag{7.138}$$

$$\{\bar{g}\}^{(k+1)} = \{\bar{r}\}^{(k)} + \alpha^{(k+1)} \{\bar{g}\}^{(k)}. \tag{7.139}$$

Значення параметра релаксації рекомендують визначати за формулами:

$$\omega = 2/(1+2\sqrt{\theta}), \text{ де } \quad \theta = (\{Z\}, \{Z\})/N; \quad \{Z\} = 0.5\{I\} + [\widetilde{L}][I], \tag{7.140}$$

а N – кількість рівнянь у СЛАР; $\{I\}$ – одиничний вектор; $[\widetilde{L}]$ – точна верхня (або нижня) нормована матриця.

Для цього варіанта алгоритму доведено умови збіжності. Кількість операцій у цьому алгоритмі приблизно така ж, як і в алгоритмі спряжених градієнтів, але швидкість його збіжності (кількість ітерацій) дещо вища.

Умови припинення описаного ітераційного процесу – друга формула (7.3).

7.5. Нормування систем лінійних алгебраїчних рівнянь

Зазвичай перед застосуванням ітераційних методів проводять нормування СЛАР. Часто це доцільно робити і для прямих методів.

Вважається, що "діагональне" нормування підвищує найменше власне значення матриці СЛАР, а оцінка для найбільшого при цьому не змінюється. Це зменшує співвідношення максимального власного значення матриці СЛАР до мінімального, яке визначає число обумовленості матриці СЛАР $\nu_{[A]}$ (див. підрозділ 5.2). Тому "діагональне" нормування призводить до більш точних розв'язків СЛАР, а також зменшує кількість ітерацій при застосуванні ітераційних методів розв'язування СЛАР (про причини цього див. п.7.1.4).

Вводиться *діагональна* матриця $[D]$ з діагональними компонентами

$$d_{mm} = \sqrt{\max_{1\le i\le N}\left(\sum_{j=1}^{N}|a_{ij}| \Big/ \sum_{j=1}^{N}|a_{mj}|\right)}, \tag{7.141}$$

де N – кількість рівнянь у СЛАР (7.1); a_{ij} – компоненти матриці $[A]$.

Спочатку обчислюються $s_i = \sum_{j=1}^{N}|a_{ij}|$, а потім $r_m = \max_{1\le i\le N}\left(s_i/s_m\right)$ та $d_{mm} = \sqrt{r_m}$.

Позначимо:

$$\{x\} = [D]\{\widetilde{x}\}. \tag{7.142}$$

Підставимо (7.142) до (7.1), помножимо результат зліва на матрицю $[D]^T = [D]$:

$$[D]^T[A][D]\{\widetilde{x}\} = [D]^T\{b\}. \tag{7.143}$$

Позначимо:

$$[\widetilde{A}] = [D]^T[A][D]; \quad \{\widetilde{b}\} = [D]^T\{b\}. \tag{7.144}$$

Отже, отримали нову (нормовану) СЛАР

$$[\widetilde{A}]\{\widetilde{x}\} = \{\widetilde{b}\}. \tag{7.145}$$

У формулі (7.144) матриця $[\widetilde{A}]$ конгруєнтна матриці $[A]$. Тому ці матриці одночасно або симетричні, або несиметричні.

Після розв'язання СЛАР (7.145) необхідно застосувати формулу (7.142) для отримання вектору $\{x\}$, який і є розв'язком СЛАР (7.1).

Але при обчисленні сум в (7.141) можливе досягнення машинної нескінченності (див п.1.3.1). Для уникнення цієї ситуації зазвичай замість $|a_{ij}|$ додають $|a_{ij}|\cdot\alpha$, де α – довільне мале число, яке доцільно обирати залежним від кількості рівнянь у СЛАР. Наприклад, $\alpha = 1/N$ або $\alpha = 1/\sqrt{N}$.

Оскільки в матриці $[D]$ компоненти $d_{mm} \geq 1$, то компоненти матриці $[\widetilde{A}]$ збільшують свої абсолютні значення. Це може призводити до випадків, коли при розв'язуванні нової СЛАР (7.145) досягається машинна нескінченність. Для уникнення цієї ситуації матрицю (7.145) зазвичай додатково нормують в такій спосіб, щоб всі діагональні елементи матриці СЛАУ дорівнювали одиниці.

Вводиться *діагональна* матриця $[W] = [\widetilde{A}_D]^{-1/2}$, тобто з компонентами

$$W_{mm} = 1/\sqrt{\widetilde{a}_{mm}}\,. \tag{7.146}$$

Дії аналогічні (7.142) … (7.145). Позначимо:

$$\{\widetilde{x}\} = [W]\{\widetilde{\widetilde{x}}\}. \tag{7.147}$$

Підставимо (7.147) до (7.145), помножимо результат зліва на матрицю $[W]^T = [W]$:

$$[W]^T[\tilde{A}][W]\{\tilde{\tilde{x}}\} = [W]^T\{\tilde{b}\}. \tag{7.148}$$

Позначимо:

$$[\widetilde{\widetilde{A}}] = [W]^T[\widetilde{A}][W]; \quad \{\widetilde{\widetilde{b}}\} = [W]^T\{\widetilde{b}\}. \tag{7.149}$$

Отже, отримали нормовану СЛАР

$$[\widetilde{\widetilde{A}}]\{\widetilde{\widetilde{x}}\} = \{\widetilde{\widetilde{b}}\}, \tag{7.150}$$

причому всі діагональні елементи матриці дорівнюють одиниці. Після розв'язання СЛАР необхідно застосувати формулу (7.147) для отримання вектору $\{\widetilde{x}\}$, який і є вектором-розв'язком СЛАР (7.145). Матриця $[\widetilde{\widetilde{A}}]$ конгруєнтна матриці $[\widetilde{A}]$, тому ці матриці одночасно або симетричні, або несиметричні.

Обідва процеси нормування СЛАР досить швидкі, оскільки мають кількість операцій, пропорційну N^2.

Примітка 7.3. Для багатьох варіантів СЛАР застосування тільки другого варіанта нормування СЛАР є достатнім для досягнення мети. Однак у випадку СЛАР з компонентами, що дуже відрізняються між собою, застосування і першого варіанта нормування СЛАР може виявитися дуже бажаним.

7.6. Завершення

7.6.1. Покращення результату розв'язування СЛАР

Є декілька способів дещо покращити вже отриманий ітераційним методом розв'язок СЛАР: Люстерніка, Гавуріна, Абрамова тощо. Зокрема у способі Люсте-

рніка застосовується мінімальне власне значення матриці $[A]$, тобто λ_1. Якщо воно відоме, то результат можна дещо покращити згідно з формулою:

$$\{x\} = \{x\}^{(k+1)} + \frac{\lambda_1}{1-\lambda_1}(\{x\}^{(k+1)} - \{x\}^{(k)}). \qquad (7.151)$$

7.6.2. Рекомендації щодо застосування методів розв'язування СЛАР

Немає єдиних рекомендацій щодо застосування тих або інших методів розв'язування СЛАР у тому або іншому випадку. Всі методи мають як недоліки, так і переваги перед іншими. Вони відмічалися при викладення методів в цьому Розділі та Розділі 5. Але можна сформулювати й *загальні* недоліки та переваги цих двох груп методів.

Основні переваги прямих методів:

- якщо матриця СЛАР повністю не вміщується в оперативну пам'ять ЕОМ, легко організувати блочний варіант роботи з нею;
- якщо СЛАР має стабільну матрицю та декілька різних правих частин СЛАР, то можна застосовувати один з прямих методів, який лише один раз проводить ті перетворення матриці СЛАР, що займають основний час для отримання розв'язку СЛАР (наприклад, за схемою Холецького).

Основні недоліки прямих методів:

- компоненти матриці $[A]$ всередині "профілю", які мали нульові значення, можуть стати ненульовими, тобто щільність заповнення матриці ненульовими компонентами підвищується (див. Примітку 5.8);
- з ростом кількості рівнянь у СЛАР росте похибка отриманого результату, навіть при добрій обумовленості СЛАР.

Основні переваги ітераційних методів:

- можна зберігати тільки ненульові компоненти матриці СЛАР, оскільки основними операціями ітераційних методів є операції перемноження матриці та вектора або двох векторів;
- самостійна корекція розв'язку, оскільки похибки, що виникають при "обрізанні" ірраціональних чисел, не накопичуються.

Основні недоліки ітераційних методів:

- для нової правої частини $\{b\}$ СЛАР (7.1) всі або майже всі дії необхідно повторювати знову, хоча при незначній зміні вектора правої частини $\{b\}$ можна скористатися раніше отриманим розв'язком як добрим наближенням для прискорення процесу отримання нового розв'язку;
- якщо матриця СЛАР повністю не вміщується в оперативну пам'ять ЕОМ, то повільну операцію зчитування блоків матриці з носія доводиться виконувати в кожній ітерації.

З простого порівняння переваг та недоліків цих двох груп методів випливає, що вони прямо протилежні. Це полегшує завдання обирання методів у конкретних ситуаціях.

Контрольні питання до підрозділу 7.1

1. Для чого були отримані чотири еквівалентні форми двошарових ітераційних схем розв'язування СЛАР?
2. Які є умови збіжності двошарових ітераційних схем?
3. Якою є загальна оцінка кількості ітерацій?
4. Які явні схеми розроблено для розв'язування СЛАР?
5. Які неявні схеми розроблено для розв'язування СЛАР?
6. Який загальний недолік є в двошарових ітераційних схемах?

Контрольні питання до підрозділу 7.2

1. Яку загальну перевагу мають ітераційні методи розв'язування СЛАР, побудовані з використанням варіаційних принципів, над методами підрозділу 7.1?
2. Які обмеження щодо властивостей матриці СЛАР мають методи мінімальних похибок наближення, мінімальних поправок та найшвидшого спуску?
3. Чому метод спряжених градієнтів вважається кращім з усіх ітераційних методів розв'язування СЛАР?

Контрольні питання до підрозділу 7.3

1. З якими методами логічно зв'язаний метод спряжених напрямків?

Контрольні питання до підрозділу 7.4

1. Які методи прискорення розроблено для отримання розв'язків СЛАР?

Контрольні питання до підрозділу 7.5

1. Для чого рекомендують проводити нормування СЛАР, та в які способи?

Контрольні питання до підрозділу 7.6

1. Чи можна покращити результати розв'язування СЛАР?
2. Що можна порекомендувати при виборі методу розв'язування СЛАР?

Розділ 8

МЕТОДИ РОЗВ'ЯЗУВАННЯ СИСТЕМ НЕЛІНІЙНИХ РІВНЯНЬ

8.1. Загальні зауваження

Всі методи розв'язування систем нелінійних рівнянь (СНР) – ітераційні.

Зазвичай розрізняють СНР загального вигляду та нелінійні САР, причому останні – як окремий випадок рівнянь загального вигляду.

Нижче елементарний випадок $N=1$ не розглядаємо (така задача називається задачею знаходження коренів (див. Розділ 3), яка є окремою великою задачею). Наводимо тільки популярні (основні) методи та алгоритми.

8.1.1. Зв'язок системи рівнянь з екстремальною задачею

Існує теорема, згідно з якою задача розв'язування системи із N рівнянь відносно N невідомих x_n (дійсні числа):

$$f_m(\{x\})=0;\quad m=1,2,...,N \tag{8.1}$$

або

$$\{F(\{x\})\}=\{f_1(\{x\}),f_2(\{x\}),...,f_N(\{x\})\}^T=\{0\}, \tag{8.2}$$

де $\{x\}=(x_1,x_2,...,x_N)^T$, еквівалентна задачі про мінімізацію функціонала

$$H(\{x\})=\sum_{m=1}^{N}[f_m(\{x\})]^2, \tag{8.3}$$

в якому $f_m(\{x\})\to 0$ – похибки наближення.

Теорема не уточнює характер рівнянь $f_m(\{x\})$: лінійні або нелінійні, алгебраїчні або трансцендентні. Однак найчастіше ця теорема застосовується при вирішенні проблем, пов'язаних з розв'язком СНР.

8.1.2. Загальні схеми ітераційних методів розв'язування систем нелінійних рівнянь

Майже всі двошарові ітераційні методи розв'язування СНР можна представити ітераційною схемою в канонічній формі:

$$[Q]^{(k+1)}\frac{\{x\}^{(k+1)}-\{x\}^{(k)}}{\tau^{(k+1)}}+\{F(\{x\})\}^{(k)}=0;\quad k=0,1,..., \tag{8.4}$$

яка є більш загальною формою схеми (7.6). Початкове (вихідне) наближення $\{x\}^{(0)}$ призначається. Якщо $[Q]$ та τ не залежать від ітерацій, то схему називають стаціонарною.

Якщо у (8.4) $[Q]^{(k+1)}=[I]$, то схема називається явною, записується у вигляді:

$$\{x\}^{(k+1)}=\{x\}^{(k)}-\tau^{(k+1)}\cdot\{F(\{x\})\}^{(k)};\quad k=0,1,..., \tag{8.5}$$

де $\tau^{(k+1)}$ визначає "розмір кроку", а $\{F(\{x\})\}^{(k)}$ – "напрямок кроку".

Параметр $\tau^{(k+1)}$ можна знаходити як такий, що мінімізує $\{F(\{x\})\}^{(k+1)}$ як функцію від $\tau^{(k+1)}$. Для цього $\{F(\{x\})\}^{(k+1)}$ апроксимують інтерполяційним многочленом або розкладають в ряд Тейлора, причому зазвичай використовують лише декілька перших членів розкладу.

Примітка 8.1. Таким методом можна знайти мінімум або максимум функції $F(\{x\})$, що задана таблицею.

Схему (8.4) можна записати як СЛАР зі змінною правою частиною:

$$[Q]^{(k+1)}\{x\}^{(k+1)} = \{g(\{x\})\}^{(k)}; \quad k = 0,1,\dots, \tag{8.6}$$

де вектор

$$\{g(\{x\})\}^{(k)} = [Q]^{(k+1)}\{x\}^{(k)} - \tau^{(k+1)}\{F(\{x\})\}^{(k)}. \tag{8.7}$$

СЛАР (8.6) для конкретної правої частини може розв'язуватися прямими або ітераційними методами. В останньому випадку такі ітерації називаються внутрішніми, а позначені номером k – зовнішніми.

Ще одна форма запису двошарових ітераційних схем розв'язування СНР:

$$\{x\}^{(k+1)} = \{S(\{x\})\}^{(k)}; \quad k = 0,1,\dots, \tag{8.8}$$

де вектор

$$\{S(\{x\})\}^{(k)} = \{x\}^{(k)} - \tau^{(k+1)}([Q]^{(k+1)})^{-1}\{F(\{x\})\}^{(k)}. \tag{8.9}$$

Доведено теорему, що розв'язок ітераційної схеми (8.8) при $k \to \infty$ прямує до точного вектор-розв'язку $\{x\}$ при будь-якому початковому $\{x\}^{(0)}$, якщо

$$\|\{S\}'\| = \|\partial\{S\} / \partial\{x\}\| \le q < 1. \tag{8.10}$$

Якщо при деякому $\{x\}_*$ вектор $\{S(\{x\}_*)\} = \{x\}_*$, то такий вектор $\{x\}_*$ називається *нерухомою точкою оператора* $\{S(\{x\})\}$. Тому, згідно з (8.8), задача знаходження розв'язку СНР (8.1) еквівалентна задачі знаходження нерухомої точки оператора $\{S(\{x\})\}$.

Оператор $\{S(\{x\})\}$ називається *стискувальним оператором,* якщо існує значення *коефіцієнта стискання* $q \in [0,1]$ таке, що для будь-яких двох векторів $\{x\}^*$ та $\{x\}^{**}$ виконується нерівність

$$\| \{S(\{x\}^*)\} - \{S(\{x\}^{**})\} \| \le q \cdot \| \{x\}^* - \{x\}^{**} \|. \tag{8.11}$$

Для оцінювання похибки розв'язку застосовується *принцип відображень, що стискують*: якщо оператор $\{S(\{x\})\}$ є визначеним на множині $\Re$ з деяким діаметром r (тобто для будь-яких векторів $\{x\}^*$ та $\{x\}^{**}$ виконується умова $\| \{x\}^* - \{x\}^{**} \| \le r$) та є таким, що стискує з коефіцієнтом стискання q, причому

$$\| \{S(\{x\}^*)\} - \{x\}^*)\} \| \le (1-q) \cdot r, \tag{8.12}$$

то оператор $\{S(\{x\})\}$ має одну нерухому точку $\{x\}_*$ і ітераційний метод (8.8) збігається до $\{x\}_*$ при будь-якому початковому $\{x\}^{(0)}$, який визначений в тій же множині $\Re$, з такими оцінками норми похибки:

$$\| \{x\}^{(k)} - \{x\}_* \| \le q^k \| \{x\}^{(0)} - \{x\}_* \| \text{ та } \| \{x\}^{(k)} - \{x\}_* \| \le \frac{q^k}{1-q} \| \{S(\{x\}^{(0)})\} - \{x\}^{(0)} \|. \tag{8.13}$$

Отже, є декілька еквівалентних форм запису ітераційних методів розв'язування СНР: (8.4), (8.6) і (8.8).

8.2. Приклади ітераційних методів розв'язування систем нелінійних рівнянь загального вигляду

8.2.1. Нелінійні методи простих ітерацій, Зейделя, Якобі

Для методу простих ітерацій (*релаксації*) $[Q]^{(k+1)} = [I]$, а $\tau^{(k+1)} = \tau$. З (8.5):

$$\{x\}^{(k+1)} = \{x\}^{(k)} - \tau \cdot \{F(\{x\})\}^{(k)}; \quad k = 0,1,\ldots. \tag{8.14}$$

Оскільки $\{S\} = \{x\} - \tau \cdot \{F(\{x\})\}$, то $\{S\}' = \partial\{S\}/\partial\{x\} = [I] - \tau \cdot [F'(\{x\})]$, де матриця

$$[F'(\{x\})] = \frac{\partial\{F(\{x\})\}}{\partial\{x\}} = \begin{bmatrix} \partial f_1(\{x\})/\partial x_1; & \partial f_1(\{x\})/\partial x_2; & \vdots & \partial f_1(\{x\})/\partial x_N \\ \partial f_2(\{x\})/\partial x_1; & \partial f_2(\{x\})/\partial x_2; & \vdots & \partial f_2(\{x\})/\partial x_N \\ \ldots & \ldots & \ldots & \ldots \\ \partial f_N(\{x\})/\partial x_1; & \partial f_N(\{x\})/\partial x_2; & \vdots & \partial f_N(\{x\})/\partial x_N \end{bmatrix}. \tag{8.15}$$

У методі *Зейделя* формула (8.14) дещо модифікується:

$$\{x\}^{(k+1)} = \{x\}^{(k)} - \tau \cdot \{F(\{x\})\}^{(k,k+1)}; \quad k = 0,1,\ldots, \tag{8.16}$$

де вектор $\{F(\{x\})\}^{(k,k+1)}$ для обчислення $(n+1)$-ї компоненти враховує вже отримані значення $x_j^{(k+1)}$, $j = 1,2,\ldots,n < N$ із вектора $\{x\}^{(k+1)}$.

Метод релаксацій та метод Зейделя мають області збіжності, які дещо відрізняються, тому мають різні значення $\tau < 1$, які забезпечують гарантовану збіжність методів. Швидкість збіжності методу Зейделя дещо вища, ніж методу релаксацій. Перевага цих методів – простота, можливість реалізації при дуже складних виразах функцій $f_m(\{x\})$.

Нелінійний метод *Якобі* має вигляд

$$f_m(x_1^{(k)}, x_2^{(k)}, \ldots, x_{j-1}^{(k)}, x_j^{(k+1)}, x_{j+1}^{(k)}, \ldots, x_N^{(k)}) = 0; \quad m = 1,2,\ldots,N, \tag{8.17}$$

причому кожне із нелінійних рівнянь (8.17) розв'язується відносно $x_j^{(k+1)}$ окремо будь-яким методом, розглянутим у Розділі 3.

8.2.2. Метод Пікара

Якщо вектор $\{F(\{x\})\}$ представлений у вигляді

$$\{F(\{x\})\} = [A]\{x\} + \{G(\{x\})\} = \{0\}, \tag{8.18}$$

де матриця $[A]$ – квадратна, то розв'язок можна знаходити за схемою:

$$[A]\{x\}^{(k+1)} + \{G(\{x\})\}^{(k)} = 0; \quad k = 0,1,\ldots. \tag{8.19}$$

Цю схему також можна записати в канонічному вигляді (8.4), де матриця розщеплення $[Q]^{(k+1)} = [A]$, а параметр $\tau^{(k+1)} = 1$.

8.2.3. Метод Ньютона-Рафсона-Канторовича

Якщо припустити, що в ітераціях $f_m(\{x\})^{(k)} \neq 0$, а $f_m(\{x\})^{(k+1)} \approx 0$, то, розкладуючи $f_m(\{x\})^{(k+1)}$ в ряд Тейлора, можна отримати, що

$$f_m(\{x\})^{(k+1)} \approx f_m(\{x\})^{(k)} + \sum_{n=1}^{N} \left(\frac{\partial f_m}{\partial x_n} \right)^{(k)} \left(x_n^{(k+1)} - x_n^{(k)} \right) = 0; \quad m = 1,2,\ldots,N; \quad k = 0,1,\ldots. \tag{8.20}$$

Цей вираз можна переписати в матричному вигляді:

$$[F'(\{x\})]^{(k)} \cdot (\{x\}^{(k+1)} - \{x\}^{(k)}) = -\{F(\{x\})\}^{(k)};\ \text{або}$$

$$\{x\}^{(k+1)} = \{x\}^{(k)} - \{F(\{x\})\}^{(k)} \cdot [\Gamma]^{(k)};\quad k = 0,1,\ldots, \tag{8.21}$$

де матриця (див. формулу (8.15))

$$[\Gamma] = [F'(\{x\})]^{-1}, \tag{8.22}$$

а вектор похибок наближення $\{F(\{x\})\}$ має компоненти $f_m(\{x\})$.

Схему (8.21) можна записати в канонічному вигляді (8.4), де матриця розщеплення $[Q]^{(k+1)} = [F'(\{x\})]^{(k+1)}$, а параметр $\tau^{(k+1)} = 1$.

Отже, в якості "розміру кроку" виступає матриця $[\Gamma]^{(k)}$, а "напрямок кроку" визначається вектором $\{F(\{x\})\}^{(k)}$ функцій похибок наближення $\{f\}^{(k)}$. Для того, щоб цей метод можна було реалізувати, необхідно довести наявність оберненої матриці $[\Gamma] = [F'(\{x\})]^{-1}$.

Л.В. Канторович довів теорему про збіжність цього методу. Згідно з цією теоремою, якщо функції $f_m(x_1, x_2, ..., x_N)$ можна двічі диференціювати і вони задовольняють умові

$$\sum_{i=1}^{N}\sum_{j=1}^{N}\left|\frac{\partial^2 f_m}{\partial x_i \partial x_j}\right| \le C;\quad m = 1,2,...,N \tag{8.23}$$

в області

$$\max_n |\, x_n - x_n^{(0)} \,| \le (1 - \sqrt{1 - 2ABC})/(AC), \tag{8.24}$$

де (обчислюються при $\{x\}^{(0)}$):

$$A \ge \max_m \sum_{n=1}^{N} |\, \Gamma_{mn} \,|;\quad B \ge \max_m \sum_{n=1}^{N} |\, \Gamma_{mn} f_n \,|;\quad C \le 1/(2AB), \tag{8.25}$$

то система (8.21) має розв'язок $\{x\}$ в тій самій області, а швидкість збіжності оцінюється як

$$\max_n |\, x_n^{(k)} - x_n^{(0)} \,| \le B \cdot (2ABC)^{2^{k-1}} / 2^{k-1}. \tag{8.26}$$

Є й інші оцінки швидкості збіжності методу, зокрема така (тут всі норми векторів – *евклідові*, а матриць – *підпорядковані* їм, див. підрозділ 4.2):

$$\| \{x\}^{(k)} - \{x\}^{(0)} \| \le M\eta q^{2^k - 1}/(1 - q^{2^k}), \tag{8.27}$$

де параметри M, η та q відповідають рівнянням

$$\| [F'(\{x\})]^{-1} \| \le M;\quad \| [F(\{x\})]^{(0)} \| \le \eta;\quad q = M^2 L \eta / 2 < 1, \tag{8.28}$$

причому

$$M\eta \sum_{k=0}^{\infty} q^{2^k - 1} < r. \tag{8.29}$$

Параметр L є параметром із умови Ліпшіца:

$$\| [F'(\{x\}^*)] - [F'(\{x\}^{**})] \| \le L \cdot \| \{x\}^* - \{x\}^{**} \|, \tag{8.30}$$

де $\{x\}^*$ та $\{x\}^{**}$ є будь-які вектори з простору радіуса r, з центром, що визначається вектором $\{x\}^{(0)}$. Така збіжність – квадратична, тобто дуже швидка.

8.2.4. Модифіковані методи Ньютона-Рафсона

Якщо обчислення матриці $[\Gamma]^{(k)}$ в ітераціях складне або неможливе, то можна застосовувати *модифікований* метод Ньютона-Рафсона:

$$\{x\}^{(k+1)} = \{x\}^{(k)} - \{F(\{x\})\}^{(k)}[\Gamma]^{(0)}; \quad k = 0,1,\ldots. \tag{8.31}$$

Цей метод має повільнішу збіжність (лінійну), може навіть не збігатися тоді, коли метод (8.21) збігається, але він простий, тому часто застосовується.

Інші модифікації методу – з параметром $\tau^{(k+1)}$:

$$[F'(\{x\})]^{(k)} \frac{\{x\}^{(k+1)} - \{x\}^{(k)}}{\tau^{(k+1)}} + \{F(\{x\})\}^{(k)} = \{0\}; \quad k = 0,1,\ldots. \tag{8.32}$$

$$[F'(\{x\})]^{(0)} \frac{\{x\}^{(k+1)} - \{x\}^{(k)}}{\tau^{(k+1)}} + \{F(\{x\})\}^{(k)} = \{0\}; \quad k = 0,1,\ldots. \tag{8.33}$$

8.2.5. Гібридні методи

Якщо матриця розщеплення $[Q]^{(k+1)} \neq [I]$, то окрім зовнішніх ітерацій (з індексом k) є і внутрішні (для "обернення" матриці $[Q]^{(k+1)}$). Популярними є такі гібридні методи:

- зовнішні ітерації – за методом Зейделя (нелінійного, п.8.2.1), внутрішні – за методом Ньютона-Рафсона-Канторовича;
- зовнішні ітерації – за методом Ньютона-Рафсона-Канторовича, внутрішні – за методом Зейделя (нелінійного, п.8.2.1);
- якщо матриця розщеплення $[Q]$ – незмінна, то внутрішні ітерації виконуються із застосуванням прямого методу з одноразовим її розкладенням на трикутні (квадратних коренів, Гаусса-Холецького, див. підрозділ 5.3).

Важливо, що у перших двох випадках у внутрішніх ітераціях використовують деяку відносно невелику фіксовану кількість ітерацій, тобто не доводять їх до збіжності. Тому такий варіант може виявитися значно вигіднішим, ніж застосування для розв'язування внутрішньої системи будь-якого *прямого* методу.

8.2.6. Градієнтні методи

В градієнтних методах зазвичай використовується схема

$$\{x\}^{(k+1)} = \{x\}^{(k)} - \tau^{(k+1)} \cdot \{H'(\{x\})\}^{(k)}; \quad k = 0,1,\ldots, \tag{8.34}$$

де $H'(\{x\}) = \partial H / \partial \{x\}$, а функціонал $H(\{x\})$ відповідає формулі (8.3).

Поширеним є варіант, коли обчислюють

$$\tau^{(k+1)} = \left(\frac{\sum_{n=1}^{N} (\partial H / \partial x_n)^2}{\sum_{m=1}^{N} \sum_{n=1}^{N} \left(\frac{\partial^2 H}{\partial x_m \partial x_n} \frac{\partial H}{\partial x_m} \frac{\partial H}{\partial x_n} \right)} \right)^{(k)}. \tag{8.35}$$

Якщо при цьому $H(\{x\})$ – аналітичний функціонал, то метод називається *методом найшвидшого спуску*. А якщо ні, тоді частинна похідна обчислюється як $\{\partial H / \partial x_m\}^{(k)} \approx \{\Delta H / \Delta x_m\}^{(k)}$, $m = 1,2,\ldots,N$ і метод називається *методом з обчисленням координат градієнта*. Швидкість збіжності градієнтних методів нижче,

ніж методу Ньютона-Рафсона-Канторовича, але є одна значна перевага: відсутність будь-яких матриць дозволяє мінімізувати об'єм пам'яті, необхідний для розв'язування задачі на ЕОМ.

8.2.7. Метод випадкових збурень

Бувають випадки, коли обчислення похідних або матриць ускладнено чи неможливе, або коли функціонал $H(\{x\})$ має декілька екстремумів. Тоді може статися у нагоді метод випадкових збурень. У цьому методі обираються випадковим чином прирости $\{\Delta x\}^{(k)} = \{\Delta x_1, \Delta x_2, ..., \Delta x_N\}^T$, обчислюються вектор $\{x\}^{(k+1)} = \{x\}^{(k)} + \{\Delta x\}$ та значення функціонала $H(\{x\})^{(k+1)}$. У залежності від того, збільшується чи зменшується $H(\{x\})^{(k+1)}$, роблять висновок про подальші $\{\Delta x\}^{(k)}$ (зміни напрямку, довжини кроку тощо).

8.3. Методи розв'язування нелінійних систем алгебраїчних рівнянь

Окремим випадком систем нелінійних рівнянь (СНР), який зустрічається досить часто, є нелінійна система *алгебраїчних* рівнянь (НСАР), яка у загальному випадку має залежну від розв'язку матрицю та праву частину

$$[A(\{x\})]\{x\} = \{b(\{x\})\}. \tag{8.36}$$

Методи розв'язування НСАР зазвичай є пристосованими методами розв'язування СНР із врахуванням таких властивостей її матриці:

- симетрична або несиметрична, позитивно визначена або ні;
- розріджена (рідко-заповнена) або ні.

8.3.1. Метод Ньютона-Рафсона

Цей метод фактично є методом Ньютона-Рафсона-Канторовича.

Вираз (8.36) записується у вигляді

$$\{\psi(\{x\})\} = \{b(\{x\})\} - [A(\{x\})]\{x\}, \tag{8.37}$$

тобто вводиться до розгляду вектор похибок наближення $\{\psi(\{x\})\}$.

Припускається, що в ітераціях $\{\psi(\{x\})\}^{(k)} \neq \{0\}$, а $\{\psi(\{x\})\}^{(k+1)} \approx \{0\}$. Позначимо $\{\Delta x\} = \{x\}^{(k+1)} - \{x\}^{(k)}$. Збереження лише перших членів розкладу похибок наближення $\{\psi(\{x\})\}^{(k+1)}$ в ряд Тейлора призводить до виразу

$$\{\psi(\{x\})\}^{(k+1)} \approx \{\psi(\{x\})\}^{(k)} + \left(\frac{\partial\{\psi(\{x\})\}}{\partial\{x\}}\right)^{(k)} \{\Delta x\} \approx \{0\}, \tag{8.38}$$

тому

$$-\left(\frac{\partial\{\psi(\{x\})\}}{\partial\{x\}}\right)^{(k)} \{\Delta x\} \approx \{\psi(\{x\})\}^{(k)}; \quad \{x\}^{(k+1)} = \{x\}^{(k)} + \{\Delta x\}. \tag{8.39}$$

Очевидно, що з урахуванням (8.37)

$$-\left(\frac{\partial\{\psi(\{x\})\}}{\partial\{x\}}\right)^{(k)} = \left([A(\{x\})] + [\widetilde{A}(\{x\})] + [\widetilde{\widetilde{A}}(\{x\})]\right)^{(k)}, \tag{8.40}$$

де введені позначення:

$$[\tilde{A}(\{x\})] = \frac{\partial\{A(\{x\})\}}{\partial\{x\}}\{x\}; \quad [\tilde{\tilde{A}}(\{x\})] = -\frac{\partial\{b(\{x\})\}}{\partial\{x\}}. \tag{8.41}$$

Остаточно отримаємо формулу ітераційного методу Ньютона-Рафсона для розв'язування нелінійної САР:

$$\left([A(\{x\})]+[\tilde{A}(\{x\})]+[\tilde{\tilde{A}}(\{x\})]\right)^{(k)}\{\Delta x\} = \{\psi(\{x\})\}^{(k)}; \quad \{x\}^{(k+1)} = \{x\}^{(k)} + \{\Delta x\}; \quad k = 0,1,.... \tag{8.42}$$

В поточній ітерації для розв'язування САР (8.42) застосовують один з методів розв'язування СЛАР: прямий або ітераційний (див. Розділи 5 та 7).

Отже, для формування СЛАР (8.42) необхідно обчислювати $[\tilde{A}(\{x\})]$ та/або $[\tilde{\tilde{A}}(\{x\})]$. Ця задача зазвичай нетривіальна. Крім того, структура заповнення матриці $[\tilde{\tilde{A}}(\{x\})]$ зазвичай не відповідає структурі заповнення матриці $[A(\{x\})]$, що має велике значення для розріджених матриць значного розміру. Тому часто нехтують вкладом матриць $[\tilde{A}(\{x\})]$ та $[\tilde{\tilde{A}}(\{x\})]$ до СЛАР (8.42), отримують *модифікований* метод Ньютона-Рафсона:

$$[A(\{x\})]^{(k)}\{\Delta x\} = \{\psi(\{x\})\}^{(k)}; \quad \{x\}^{(k+1)} = \{x\}^{(k)} + \{\Delta x\}; \quad k = 0,1,.... \tag{8.43}$$

Він збігається значно повільніше, може зовсім не збігатися.

Якщо нехтування матрицями $[\tilde{A}(\{x\})]$ або $[\tilde{\tilde{A}}(\{x\})]$ вкрай небажано, але їх точне обчислення неможливе, то застосовують наближені формули диференціювання, які потребують збереження попередніх матриць та векторів (в індексній формі запису):

$$\tilde{A}_{mn} = \frac{\partial A_{mn}(x_j)}{\partial x_i}x_i \approx \frac{A_{mn}^{(k)} - A_{mn}^{(k-1)}}{x_i^{(k)} - x_i^{(k-1)}}x_i^{(k)}; \quad \tilde{\tilde{A}}_{mn} = -\frac{\partial b_m(x_j)}{\partial x_n} \approx -\frac{b_m^{(k)} - b_m^{(k-1)}}{x_n^{(k)} - x_n^{(k-1)}}. \tag{8.44}$$

8.3.2. Метод простих ітерацій та метод Зейделя

У методі простих ітерацій НСАР (8.36) записується як ітераційний процес:

$$[A(\{x\})]^{(k)}\{x\}^{(k+1)} = \{b(\{x\})\}^{(k)}. \tag{8.45}$$

У методі Зейделя

$$[A(\{x\})]^{(k,k+1)}\{x\}^{(k+1)} = \{b(\{x\})\}^{(k,k+1)}, \tag{8.46}$$

де матриця $[A(\{x\})]^{(k,k+1)}$ та вектор $\{b(\{x\})\}^{(k,k+1)}$ для обчислення компоненти $x_{n+1}^{(k+1)}$ враховує вже отримані значення $x_j^{(k+1)}$; $j = 1,2,...,n < N$ із вектора $\{x\}^{(k+1)}$. Таку процедуру в нелінійних САР не завжди можна реалізувати.

Аналогічно модифікуються й інші методи розв'язування СНР для отримання розв'язків нелінійних САР.

Контрольні питання до підрозділу 8.1

1. Як формулюється зв'язок системи рівнянь з екстремальною задачею?

Контрольні питання до підрозділу 8.2

1. Які загальні риси та відмінності є у методів релаксації, Зейделя, Якобі?
2. Яку збіжність має метод Ньютона-Рафсона-Канторовича?

3. Чим градієнтні методи принципово відрізняються від методу Ньютона-Рафсона-Канторовича?

Контрольні питання до підрозділу 8.3

1. Чим метод Ньютона-Рафсона відрізняється від методу Ньютона-Рафсона-Канторовича?

2. Як формулюються метод простих ітерацій та метод Зейделя для розв’язання системи нелінійних алгебраїчних рівнянь?

Розділ 9

ЧИСЕЛЬНЕ ІНТЕРПОЛЮВАННЯ, НАБЛИЖЕННЯ ТА ДИФЕРЕНЦІЮВАННЯ ФУНКЦІЙ

9.1. Загальні зауваження

Якщо на основі таблиці $y_i = f(z_i)$ знаходять деяку безперервну аналітичну функцію $\overline{f}(z)$, яка заздалегідь є невідомою, то говорять про *відновлювання* функції $f(z)$, що задана таблицею. Всі значення аргументу z_i (вузли) належать деякому інтервалу $[\alpha, \beta]$, тобто $z_i \in [\alpha, \beta]$, причому $\alpha = z_0 < z_1 < ... < z_{N-1} < z_N = \beta$ (зокрема, *немає вузлів, що співпадають*). Обов'язковою умовою відновлення є умова $\overline{f}(z_i) = y_i = f(z_i)$. Інакше кажучи, значення відновленої функції у вузлах повинні дорівнювати табличним.

Мета відновлювання може бути різною. Наприклад: обчислити значення функції в іншій точці ("ущільнити" таблицю); провести диференціювання функції в заданій точці; знайти, яке або які значення має аргумент для конкретного значення функції (оборотне відновлювання), знайти корені функції, інші.

Інтерполюванням називають відновлення функції $f(z)$ для подальшого застосування $\overline{f}(z)$ лише при $z \in [\alpha, \beta]$, а *екстраполюванням* – за межами інтервалу $[\alpha, \beta]$. При екстраполюванні існує більш значний ризик одержати невірний результат, ніж при інтерполюванні, оскільки за межами інтервалу $[\alpha, \beta]$ інформація про реальну поведінку функції $f(z)$ зазвичай відсутня повністю або частково (навіть такі, як знак функції, її асимптота, знаки похідних, інші).

Наближенням безперервної функції $f(z)$ називають знаходження деякої нової функції $\overline{f}(z)$ з областю визначення $[\alpha, \beta]$, яка з достатньою точністю замінює функцію $f(z)$. Умови наближення можуть бути різними, наприклад: точно відповідати вузловим значенням $y_i = f(z_i)$ та зі вказаною точністю наближувати функцію $f(z)$ проміж вузлів; мати безперервні похідні до визначеного порядку включно; мати на кінцях діапазону або у вказаних вузлах конкретні значення похідних; за енергією (інтегрально), інші.

Наближення застосовують для заміни складної функції $f(z)$ на іншу просту, або комбінацію інших простих, наприклад, для спрощення її обчислення (а в ЕОМ тригонометричні функції, логарифми та інші "неалгебраїчні" функції інакше обчислити не можна) або для подальшого її наближеного інтегрування (див. Розділ 10).

Ця тема – дуже значна за обсягом. Функції можуть мати два та більшу кількість аргументів. Крім того, ця тема має суміжні теми. Тут розглянемо функції тільки одного аргументу та лише основні методи і формули.

Перш ніж розглядати інтерполювання функцій, необхідно ввести поняття скінченних різниць. Суттєве таке: в таблиці $y_i = f(z_i)$ значення аргументу може змінюватися або з однаковим кроком, або з різним. У підрозділах 9.2 ... 9.4 розглянуто перший випадок, у 9.5 – другий.

9.2. Скінченні різниці різних порядків та таблиці різниць

Позначимо деяку функцію як $y = f(z)$, а також $h = \Delta z$ – приріст її аргументу (крок зміни аргументу). Тоді позначимо прирощення функції:

$$\begin{aligned} &\Delta y = \Delta f(z) = f(z+h) - f(z); \\ &\Delta^{(2)} y = \Delta(\Delta y) = \Delta f(z+h) - \Delta f(z); \\ &\ldots\ldots\ldots\ldots \\ &\Delta^{(n)} y = \Delta(\Delta^{(n-1)} y). \end{aligned} \tag{9.1}$$

Можна отримати, що

$$\Delta^{(n)} f(z) = f(z+nh) - C_n^1 f(z+(n-1)h) + C_n^2 f(z+(n-2)h) - \ldots + (-1)^n f(z), \tag{9.2}$$

де коефіцієнти

$$C_n^m = \frac{n(n-1)\cdot\ldots\cdot[n-(m-1)]}{m!}. \tag{9.3}$$

Доведено **теорему**: якщо $y = f(z)$ має безперервну похідну $f^{(n)}(z)$ на відрізку $[z, z+nh]$, то $\Delta^{(n)} f(z) = h^n f^{(n)}(z+\Theta nh)$ при $0 < \Theta < 1$.

Наслідок. $f^{(n)}(z+\Theta nh) = \Delta^{(n)} f(z)/h^n$ та $\lim\limits_{h\to 0} f^{(n)}(z+\Theta nh) = f^{(n)}(z)$, тобто при малих h похідна

$$f^{(n)}(z) \approx \frac{\Delta^{(n)} f(z)}{h^n}. \tag{9.4}$$

Відповідно до (9.1) можна записати, що для i-ої точки:

$$\begin{aligned} &\Delta y_i = y_{i+1} - y_i; \\ &\Delta^{(2)} y_i = \Delta(\Delta y_i) = \Delta y_{i+1} - \Delta y_i; \\ &\ldots\ldots\ldots \\ &\Delta^{(n)} y_i = \Delta(\Delta^{(n-1)} y_i) = \Delta^{(n-1)} y_{n+1} - \Delta^{(n-1)} y_i. \end{aligned} \tag{9.5}$$

Вираз (9.2) одержує вигляд

$$\Delta^{(n)} y_i = y_{n+i} - C_n^1 y_{n+i-1} + C_n^2 y_{n+i-2} - \ldots + (-1)^n y_i. \tag{9.6}$$

Приклад 9.1. При $n = 2$: $\Delta^{(2)} y_i = y_{i+2} - 2y_{i+1} + y_i$; при $n = 3$: $\Delta^{(3)} y_i = y_{i+3} - 3y_{i+2} + 3y_{i+1} - y_i$; при $n = 4$: $\Delta^{(4)} y_i = y_{i+4} - 4y_{i+3} + 6y_{i+2} - 4y_{i+1} + y_i$.

Використовуючи наближену формулу (9.4), отримаємо, що:

$y_i''(z) \approx \Delta^{(2)} y_i / h^2 = (y_{i+2} - 2y_{i+1} + y_i)/h^2$;

$y_i'''(z) \approx \Delta^{(3)} y_i / h^3 = (y_{i+3} - 3y_{i+2} + 3y_{i+1} - y_i)/h^3$;

$y_i^{IV}(z) \approx \Delta^{(4)} y_i / h^4 = (y_{i+4} - 4y_{i+3} + 6y_{i+2} - 4y_{i+1} + y_i)/h^4$.

Ці формули застосовуються у методі скінченних різниць для апроксимації похідних другого, третього та четвертого порядків.

З різниць (9.5) можна створити *таблиці різниць*. Використовують *діагональну* та *горизонтальну* таблиці різниць. Зокрема, при $n=3$ вони мають вигляд:

z	y	$\Delta^{(1)}y$	$\Delta^{(2)}y$	$\Delta^{(3)}y$
z_0	y_0			
		Δy_0		
z_1	y_1		$\Delta^{(2)}y_0$	
		Δy_1		$\Delta^{(3)}y_0$
z_2	y_2		$\Delta^{(2)}y_1$	
		Δy_2		
z_3	y_3			

z	y	$\Delta^{1}y$	$\Delta^{(2)}y$	$\Delta^{(3)}y$
z_0	y_0	Δy_0	$\Delta^{(2)}y_0$	$\Delta^{(3)}y_0$
z_1	y_1	Δy_1	$\Delta^{(2)}y_1$	
z_2	y_2	Δy_2		
z_3	y_3			

Приклад 9.2. Функція $y=f(z)=3z^3-5z^2-z+3$. Діагональна таблиця різниць при $z_0=0$ та $h=1$:

z	y	$\Delta^{(1)}y$	$\Delta^{(2)}y$	$\Delta^{(3)}y$	$\Delta^{4}y$
0	3				
		-3			
1	0		8		
		5		18	
2	5		26		0
		31		18	
3	36		44		
		75			
4	111				

Примітка 9.1. Якщо при обчислюванні значень для таблиць різниць десь зроблено помилку, то при розрахунках наступних різниць похибка у значеннях тільки *збільшується*.

9.3. Інтерполяційні формули Ньютона

Для інтерполювання табличної функції одного аргументу будується аналітична функція $y=\overline{f}(z)$ така, що максимально точно апроксимує цю функцію в околі заданої точки. При цьому виконують умову $\overline{f}(z_i)=y_i=f(z_i)$.

Відомо, що через $(N+1)$ вузол можна провести безліч ліній. Але якщо застосувати *поліном n-ої степені* при $n=N$, то він опише лише одну лінію, тобто з'являється однозначність. Тому усі інтерполяційні формули, які побудовані на основі поліномів, мають n-ю степінь (порядок апроксимації) та $(N+1)$ членів, спираються на $(N+1)$ значень таблиці в діапазоні $z_i\in[\alpha,\ \beta]$. Зазвичай такі формули пишуться у загальному вигляді (значення $n=N$ не оговорено), а

застосовується стільки членів, скільки можливо або скільки потрібно для одержання достатньої точності.

Введемо позначення:

$$q = (z - z_0)/h. \tag{9.7}$$

Тут z_0 – початкова точка, або початкове значення аргументу.

9.3.1. Перша інтерполяційна формула Ньютона

Перша інтерполяційна формула Ньютона має вигляд:

$$P_n(z) \approx y_0 + q\Delta y_0 + \frac{q(q-1)}{2!}\Delta^{(2)}y_0 + \ldots + \frac{q(q-1)\cdot\ldots\cdot(q-n+1)}{n!}\Delta^{(n)}y_0. \tag{9.8}$$

При $n=1$ – формула лінійного інтерполювання: $P_1(z) = y_0 + q\Delta y_0$; при $n=2$ – формула квадратичного інтерполювання: $P_2(z) \approx y_0 + q\Delta y_0 + [q(q-1)/2]\Delta^{(2)}y_0$.

Застосовується в околі x_0, тобто від початку таблиці (інтерполювання вперед або екстраполювання назад), у межах одного кроку, тобто при $|q| \le 1$.

Формули для оцінки похибок інтерполяційних формул зазвичай утворюють, припускаючи, що $\Delta^{(n+1)}y = const$.

Формула для похибки 1-ої інтерполяційної формули Ньютона:

$$R_n(z) \approx \frac{q(q-1)\ldots(q-n)}{(n+1)!}\Delta^{(n+1)}y_0, \text{ якщо } \Delta^{(n+1)}y = const. \tag{9.9}$$

9.3.2. Друга інтерполяційна формула Ньютона

Друга інтерполяційна формула Ньютона має вигляд:

$$P_n(z) = y_n + q\Delta y_{n-1} + \frac{q(q+1)}{2!}\Delta^{(2)}y_{n-2} + \ldots + \frac{q(q+1)\cdot\ldots\cdot(q+n-1)}{n!}\Delta^{(n)}y_0. \tag{9.10}$$

Застосовується в околі x_n, тобто в кінці таблиці (інтерполювання назад або екстраполювання вперед), у межах одного кроку, тобто при значеннях $|q| \le 1$.

Формула для похибки 2-ої інтерполяційної формули Ньютона має вигляд

$$R_n(z) \approx \frac{q(q+1)\ldots(q+n)}{(n+1)!}\Delta^{(n+1)}y_0, \text{ якщо } \Delta^{(n+1)}y = const. \tag{9.11}$$

Приклад 9.3. Потрібно визначити $\sin 14^o$, якщо є значення функції для 15, 20, 25 та 30 градусів. Будуємо, наприклад, горизонтальну таблицю різниць:

z	$y = \sin z$	$\Delta^{(1)}y$	$\Delta^{(2)}y$	$\Delta^{(3)}y$
15	0.2588	0.0832	-0.0026	-0.0006
20	0.3420	0.0806	-0.0032	-
25	0.4226	0.0774	-	-
30	0.5000	-	-	-

Визначимо: $q = (14-15)/5 = -0.2$. Тоді, застосовуючи першу формулу Ньютона та проводячи екстраполяцію назад, отримаємо, що:

$$\sin 14^o \approx 0.2588 + (-0.2)\cdot 0.0823 + [(-0.2)(-0.2-1)/2](-0.0026) +$$
$$+[(-0.2)(-0.2-1)(-0.2-2)/6](-0.0006) = 0.2419.$$

В даному випадку всі цифри – вірні, оскільки більш точне значення $\sin 14^o \approx 0.241922...$.

9.4. Центрально-різницеві інтерполяційні формули

Перша та друга інтерполяційні формули Ньютона – односторонні, тобто мають одне направлення. Але на їх основі можна побудувати центрально-різницеві інтерполяційні формули, які опираються на окіл вказаної точки в обох напрямах та "працюють" від неї теж в обох напрямах.

Якщо у діагональній таблиці різниць перенумерувати точки "від центру", отримаємо таблицю центральних різниць:

x	y	$\Delta^1 y$	$\Delta^2 y$	$\Delta^3 y$	$\Delta^4 y$
x_{-2}	y_{-2}				
		$\Delta^1 y_{-2}$			
x_{-1}	y_{-1}		$\Delta^2 y_{-2}$		
		$\Delta^1 y_{-1}$		$\Delta^3 y_{-2}$	
x_0	y_0		$\Delta^2 y_{-1}$		$\Delta^4 y_{-2}$
		$\Delta^1 y_0$		$\Delta^3 y_{-1}$	
x_{-1}	y_1		$\Delta^2 y_0$		
		$\Delta^1 y_1$			
x_{-2}	y_2				

Центрально-різницевими формулами є формули Гаусса, Стірлінга та Бесселя. Вони застосовують $2n+1$ вузлів, між якими рівні відстані.

9.4.1. Інтерполяційні формули Гаусса

Перша інтерполяційна формула Гаусса:

$$P_n(z) = y_0 + q\Delta y_0 + \frac{q(q-1)}{2!}\Delta^{(2)} y_{-1} + \frac{(q+1)(q-1)}{3!}\Delta^{(3)} y_{-1} + \frac{(q+1)q(q-1)(q-2)}{4!}\Delta^{(4)} y_{-2} +$$
$$+ \frac{(q+2)(q+1)q(q-1)(q-2)}{5!}\Delta^{(5)} y_{-2} + ...$$
$$+ \frac{(q+n-1)...(q-n+1)}{(2n-1)!}\Delta^{(2n-1)} y_{-(n-1)} + \frac{(q+n-1)...(q-n)}{(2n)!}\Delta^{(2n)} y_{-n}, \qquad (9.12)$$

де q знов відповідає формулі (9.7).

Друга інтерполяційна формула Гаусса:

$$P_n(z) = y_0 + q\Delta y_{-1} + \frac{(q+1)q}{2!}\Delta^{(2)} y_{-1} + \frac{(q+1)q(q-1)}{3!}\Delta^{(3)} y_{-2} + \frac{(q+2)(q+1)q(q-1)}{4!}\Delta^{(4)} y_{-2} + ...$$
$$+ \frac{(q+n-1)...(q-n+1)}{(2n-1)!}\Delta^{(2n-1)} y_{-n} + \frac{(q+n)(q+n-1)...(q-n+1)}{(2n)!}\Delta^{(2n)} y_{-n}. \qquad (9.13)$$

Формули Гаусса застосовують рідко, оскільки є дві інші: Стірлінга та Бесселя, які потребують менший обсяг обчислень та є більш точними.

9.4.2. Інтерполяційна формула Стірлінга

Ця формула – середнє арифметичне 1-ої та 2-ої формули Гаусса:

$$P_n(z) = y_0 + q\frac{\Delta y_{-1} + \Delta y_0}{2} + \frac{q^2}{2!}\Delta^{(2)} y_{-1} + \frac{q(q^2-1^2)}{3!}\frac{\Delta^{(3)} y_{-2} + \Delta^{(3)} y_{-1}}{3} + \ldots. \quad (9.14)$$

Формула для похибки також утворюється як середнє арифметичне похибок 1-ої та 2-ої формули Гаусса:

$$R_n(z) \approx \frac{\Delta^{(2n+1)} y_{-n-1} + \Delta^{(2n+1)} y_{-n}}{2(2n+1)!} q(q^2-1^2)(q^2-2^2)\ldots(q^2-n^2), \text{ якщо } \Delta^{(n+1)} y = const, \quad (9.15)$$

де $2n$ – порядок максимальної використаної різниці.

Рекомендують застосовувати при $|q| \le 0.5$.

9.4.3. Інтерполяційна формула Бесселя

Виводиться із застосуванням 2-ої формули Гаусса. Спочатку за середню точку беруться почергово дві сусідні, потім від одержаних двох виразів утворюється середнє:

$$P_n(z) = \frac{y_0 + y_1}{2} + \left(q - \frac{1}{2}\right)\Delta y_0 + \frac{q(q-1)}{2}\frac{\Delta^{(2)} y_{-1} + \Delta^{(2)} y_0}{2} + \frac{(q-1/2)q(q-1)}{3!}\Delta^{(3)} y_{-1} + \ldots. \quad (9.16)$$

При $q = 1/2$ формула значно спрощується, називається формулою інтерполювання "на середину". Іноді формулу Бесселя записують через $p = q - 1/2$. Тоді вона приймає симетричний вигляд.

Формула для похибки утворюється аналогічним шляхом та має вигляд:

$$R_n(z) \approx \frac{\Delta^{(2n+2)} y_{-n-1} + \Delta^{(2n+2)} y_{-n}}{2(2n+2)!} q(q^2-1^2)(q^2-2^2)\ldots(q^2-n^2)[q-(n+1)], \text{ якщо } \Delta^{(n+1)} y = const, \quad (9.17)$$

де $2n+1$ – порядок максимальної використаної різниці.

Рекомендують застосовувати при $0.25 \le |q| \le 0.75$.

9.5. Інтерполяційні формули для різно-віддалених вузлів

Випадок, коли таблична функція має значення z_i з різними відстанями (має різні кроки аргументу) – звичайний.

9.5.1. Інтерполяційна формула Лагранжа

Нехай є $(N+1)$ "опорних" вузлів таблиці, тобто маємо $y_i = f(z_i)$, $i = 0,1,\ldots,N$. Інтерполяційна формула Лагранжа має вигляд:

$$\Lambda_n(z) = \sum_{i=0}^{n} \frac{(z-z_0)(z-z_1)\ldots(z-z_{i-1})(z-z_{i+1})\ldots(z-z_n)}{(z_i-z_0)(z_i-z_1)\ldots(z_i-z_{i-1})(z_i-z_{i+1})\ldots(z_i-z_n)} \cdot y_i = \sum_{i=0}^{n} \Lambda_i^n \cdot y_i, \quad (9.18)$$

де $n = N$. Якщо позначити:

$$\Pi_{n+1}(z) = (z-z_0)(z-z_1)\ldots(z-z_n), \quad (9.19)$$

то можна отримати інший запис цієї формули:

$$\Lambda_n(z) = \Pi_{n+1}(z) \cdot \sum_{i=0}^{n} \frac{y_i}{\Pi'_{n+1}(z_i)\cdot(z-z_i)}. \quad (9.20)$$

Примітка 9.2. Вигляд формул не зміниться, якщо провести заміну $z = At + B$, зміниться тільки аргумент: з z на t.

Примітка 9.3. Іноді у застосуванні дуже важливо те, що інтерполяційна формула Лагранжа в явному вигляді містить величини y_i.

Примітка 9.4. Очевидно, що формула Лагранжа може застосовуватися й у випадку рівномірних кроків.

Похибка інтерполяційної формули Лагранжа (9.18) обчислюється за формулою

$$R_n(z) = \frac{f^{(n+1)}(\xi)}{(n+1)!} \cdot \Pi_{n+1}(z), \tag{9.21}$$

де $\xi \in [\alpha, \beta]$ залежить від z; або

$$\left|R_n(z)\right| \le \frac{M_{n+1}}{(n+1)!} \cdot \left|\Pi_{n+1}(z)\right|, \tag{9.22}$$

де $M_{n+1} = \max\left|f^{(n+1)}(\xi)\right|$ при $\xi \in [\alpha, \beta]$.

Оскільки значення $\left|\Pi_{n+1}(z)\right|$ визначається тільки розташуванням вузлів, то похибка також залежить від цього.

П.Л. Чебишев довів, що найкраще розташування вузлів (якщо це можливо робити) визначається формулою:

$$z_i = [(\beta+\alpha) + (\beta-\alpha)\xi_i]/2, \quad \text{де} \quad \xi_i = -\cos\left(\frac{2i+1}{2n+2} \cdot \pi\right); \quad i = 0,1,...,n. \tag{9.23}$$

Ці вузли відповідають нулям полінома Чебишева $T_{n+1}(z)$. Вони згущаються біля кінців відрізка. В такому разі $\left|\Pi_{n+1}(z)\right| \le 2[(\beta-\alpha)/4]^{n+1}$.

Приклад 9.4. При $n=1$ отримаємо рівняння прямої, що проходить через точки α та β:

$$y = \frac{z-\beta}{\alpha-\beta} y_0 + \frac{z-\alpha}{\beta-\alpha} y_1.$$

При $n=2$ отримаємо рівняння параболи, що проходить через точки α, β та γ: $y = \dfrac{(z-\beta)(z-\gamma)}{(\alpha-\beta)(\alpha-\gamma)} y_0 + \dfrac{(z-\alpha)(z-\gamma)}{(\beta-\alpha)(\beta-\gamma)} y_1 + \dfrac{(z-\alpha)(z-\beta)}{(\gamma-\alpha)(\gamma-\beta)} y_2$.

Приклад 9.5. Завдання: знайти значення функції $y = \sin(\pi z)$ при $z = 0.2$.

Ще зі шкільної програми запам'яталося, що $y_0 = \sin(0) = 0$, $y_1 = \sin(\pi/6) = 1/2$, $y_2 = \sin(\pi/2) = 1$. Тому спочатку отримаємо наближення функції $y = \sin(\pi z)$ на основі трьох вузлів: $\alpha = z_0 = 0$, $\beta = z_1 = 1/6$, $\gamma = z_2 = 1/2$. Оскільки вузлів – три, то $n = 2$. Згідно з формулою параболи (див. приклад 9.4) запишемо:

$$y(z) \approx L_2(z) = \frac{(z-1/6)(z-1/2)}{(0-1/6)(0-1/2)} \cdot 0 + \frac{(z-0)(z-1/2)}{(1/6-0)(1/6-1/2)} \cdot \frac{1}{2} + \frac{(z-0)(z-1/6)}{(1/2-0)(1/2-1/6)} \cdot 1 = 3.5z - 3z^2.$$

Ця формула при $z = 0.2$ дає інтерпольоване значення $y(\pi \cdot 0.2) \approx$ $\approx 3.5 \cdot 0.2 - 3 \cdot 0.2^2 = 0.58$. Більш точне значення $y(\pi \cdot 0.2) \approx 0.587785...$. Тобто

результат такого інтерполювання для точки $z = 0.2$ має відносну похибку $\delta \approx [(0.587785 - 0.58)/0.587785] \cdot 100\% \approx 1.32\%$ (в інших точках ця похибка буде іншою). Така досить велика похибка інтерполювання викликана значною відстанню між вузлами апроксимації та малою кількістю вузлів.

9.5.2. Інтерполяційна формула Ньютона

Інтерполяційна формула Ньютона має вираз

$$P_n(z) = y_0 + \Delta\tilde{y}_0^{(1)}(z - z_0) + \Delta\tilde{y}_0^{(2)}(z - z_0)(z - z_1) + ... + \Delta\tilde{y}_0^{(n)}(z - z_0)(z - z_1)...(z - z_{n-1}), \quad (9.24)$$

де позначені так звані *поділені різниці* (від слова ділити):

$$\begin{cases} \Delta\tilde{y}_0^{(1)} = (y_1 - y_0)/(z_1 - z_0); & ... & \Delta\tilde{y}_k^{(1)} = (y_{k+1} - y_k)/(z_{k+1} - z_k); & ... \\ \Delta\tilde{y}_0^{(2)} = (\Delta\tilde{y}_1^{(1)} - \Delta\tilde{y}_0^{(1)})/(z_2 - z_0); & ... & \Delta\tilde{y}_k^{(2)} = (\Delta\tilde{y}_{k+1}^{(1)} - \Delta\tilde{y}_k^{(1)})/(z_{k+2} - z_k); & ... \\ ... & ... & ... & ... \\ \Delta\tilde{y}_0^{(m)} = (\Delta\tilde{y}_1^{(m-1)} - \Delta\tilde{y}_0^{(m-1)})/(z_m - z_0); & ... & \Delta\tilde{y}_k^{(m)} = (\Delta\tilde{y}_{k+1}^{(m-1)} - \Delta\tilde{y}_k^{(m-1)})/(z_{k+m} - z_k); & ... \\ ... & ... & ... & ... \end{cases} \quad (9.25)$$

де $k = 0,1,...$, $m = 2,3,...$ Їх можна збирати до таблиць (поділених) різниць.

Є **лема**: якщо функція $y(z) = P_n(z)$, тобто вона є поліномом n-ої степені, то його поділена різниця $(n+1)$-го порядку дорівнює нулю.

Приклад 9.6. Знайти інтерпольоване значення функції $y(z)$ (див. перші два стовпці таблиці) для $z = 3.7608$.

z	$y(z)$	$\Delta^{(1)}y$	$\Delta^{(2)}y$	$\Delta^{(3)}y$
0	0.3989423	-0.0000500	-0.0000199	0
2.5069	0.3988169	-0.0001499	-0.0000199	-
5.0154	0.3984408	-0.0002496	-	-
7.5270	0.3978138	-	-	-

Отже, із застосуванням формули (9.24) отримаємо інтерполяційний поліном: $y(z) = 0.3989423 - 0.0000500z - 0.0000199z(z - 2.5069)$. При $z = 3.7608$ одержимо, що $y(3.7608) = 0.3989423 - 0.0000500 \cdot 3.7608 - 0.0000199 \cdot 3.7608 \cdot \cdot(3.7608 - 2.5069) \approx 0.3986604$.

9.6. Інтерполювання кубічними сплайнами

Суттєвим недоліком інтерполювання розглянутими вище поліномами Лагранжа, Ньютона та іншими є те, що при збільшенні кількості вузлів інтерполяції в діапазоні $[\alpha, \beta]$ громіздкість формул росте, а похибка інтерполювання може не тільки зменшуватися, а й збільшуватися, навіть до нескінченності. Наприклад, це виявилося при інтерполюванні поліномами Лагранжа на рівномірній сітці функцій $f(z) = (1 + 25z^2)^{-1}$ (Рунге, 1901 р.) та $f(z) = |z|$ (Бернштейн, 1916 р.) у діапазоні $z \in [-1,1]$.

Тому замість інтерполювання поліномами рекомендують застосовувати сплайни. Щодо походження назви. Spline (рейка) – креслярський інструмент з гнучкою металевою лінійкою, поставленою на ребро, за допомогою якої на кресленні можна проводити через декілька точок плавні криві.

Тема сплайн-апроксимацій дуже значна (див., наприклад, книгу [20]), тому тут розглянемо лише основні положення цього методу.

Якщо для безперервної функції $f(z)$ обрати вузли $z_i \in [\alpha, \beta]$ такі, що $\alpha = z_0 < z_1 < ... < z_{N-1} < z_N = \beta$, то одержимо таблицю $y_i = f(z_i)$, де $i = 0,1,...,N$.

Сплайн – *кусково-многочленна* функція $s(z)$ *невисокої* степені n, яка в діапазоні $z \in [\alpha, \beta]$ інтерполює $f(z)$ через значення $y_i = f(z_i)$ і має безперервні похідні до p-ої включно (тобто відноситься до класу C^p). Число $k = n - p$ називають дефектом сплайну. Тобто функція $s(z)$ створена з деякої кількості "кусків" на основі многочленів, причому ці куски "зшити" між собою так, щоб забезпечити безперервність і самої функції, і p її перших похідних.

Найбільш популярними у застосуванні є кубічні ($n = 3$) сплайни класу $s(z) \in C^2[\alpha, \beta]$, тобто з дефектом $k = 1$.

Кубічний сплайн класу C^2 з дефектом $k = 1$ має такі властивості:

а/ на кожному сегменті $[z_{i-1}, z_i]$, де $i = 1,2...,N$ є многочленом 3-ої степені;

б/ $s(z)$, $s'(z)$, $s''(z)$ безперервні в діапазоні $[\alpha, \beta]$;

в/ $s(z_i) = f(z_i) = y_i$, де $i = 0,1,...,N$ (умова інтерполювання).

Існує чотири форми представлення сплайнів: як сума усічених степеневих функцій, через фундаментальні сплайни, кусково-многочленна, через *B*-сплайни. Оскільки перша форма призводить до швидкого накопичення похибок, а друга – до значної кількості дій, то ці дві форми на практиці зазвичай не застосовуються, хоча бувають зручнішими при теоретичних викладках.

Використаємо третю форму, а саме кусково-многочленну форму представлення кубічного сплайну.

Для побудови кубічного сплайну $s(z)$ розглядають кожний i-й сегмент $[z_{i-1}, z_i] \in [\alpha, \beta]$, в якому функцію $s_i(z)$ шукають у вигляді розкладу в ряд

$$s_i(z) = a_i + b_i(z - z_i) + \frac{c_i}{2}(z - z_i)^2 + \frac{d_i}{6}(z - z_i)^3; \quad z \in [z_{i-1}, z_i]; \quad i = 1,2...,N. \qquad (9.26)$$

З (9.26) та в/ випливає, що $a_i = s_i(z_i) = f(z_i) = y_i$; $b_i = s_i'(z_i) = y_i'$; $c_i = s_i''(z_i) = y_i''$; $d_i = s_i'''(z_i) = y_i'''$. Додатково приймають, що $a_0 = f(\alpha) = y_\alpha$.

Перша з умов безперервності (див. пункт б/) потребує, щоб $s_i(z_i) = s_{i+1}(z_i)$, $i = 1,2,...,N-1$. Тому, з урахуванням (9.26):

$$a_i = a_{i+1} + b_{i+1}(z_i - z_{i+1}) + \frac{c_{i+1}}{2}(z_i - z_{i+1})^2 + \frac{d_{i+1}}{6}(z_i - z_{i+1})^3; \quad i = 1,2,...,N-1. \qquad (9.27)$$

Введемо позначення: $h_{i+1} = z_{i+1} - z_i$. Оскільки $a_i = y_i$, то з (9.27) спочатку утворюється вираз

$$h_{i+1}b_{i+1} - \frac{h_{i+1}^2}{2}c_{i+1} + \frac{h_{i+1}^3}{6}d_{i+1} = y_{i+1} - y_i; \quad i = 1,2,...,N-1, \qquad (9.28)$$

а потім, зменшенням індексів на одиницю

$$h_i b_i - \frac{h_i^2}{2} c_i + \frac{h_i^3}{6} d_i = y_i - y_{i-1}; \quad i = 2,3,...,N. \tag{9.29}$$

За умови безперервності перших похідних (див. пункт б/ властивостей) у всіх внутрішніх вузлах, тобто при $i = 1,2,...,N-1$ повинно бути $s_i'(z_i) = s_{i+1}'(z_i)$. Аналогічно попередньому можна отримати, що:

$$h_i c_i - \frac{h_i^2}{2} d_i = b_i - b_{i-1}; \quad i = 2,3...,N. \tag{9.30}$$

А умова безперервності других похідних у всіх внутрішніх вузлах $s_i''(z_i) = s_{i+1}''(z_i)$, $i = 1,2,...,N-1$ призводить до рівнянь

$$h_i d_i = c_i - c_{i-1}; \quad i = 2,3,...,N. \tag{9.31}$$

Всього потрібно мати значення $4N$ коефіцієнтів. Вже відомі $N+1$ значень $a_i = y_i$, а також є СЛАР із $3(N-1)$ рівнянь (9.29) ... (9.31) відносно $3N$ коєфіцієнтів b_i, c_i, d_i при $i = 2,3,...,N$. Додаткові вирази отримують завдяки граничним умовам (ГУ). Існує багато варіантів ГУ, найпопулярніші:

№ варіанта	Граничні умови
I	$s'(\alpha) = f'(\alpha) = y'_\alpha; \quad s'(\beta) = f'(\beta) = y'_\beta$
II	$s''(\alpha) = f''(\alpha) = y''_\alpha; \quad s''(\beta) = f''(\beta) = y''_\beta$
III	$s^{(n)}(\alpha) = s^{(n)}(\beta); \quad n = 0,1,2$ (періодичність, з періодом $\beta - \alpha$)
IV	$s'''(z_m + 0) = f'''(z_m - 0); \quad m = 1,2,...,N-1$

Для отримання більш прийнятного вигляду СЛАР зазвичай з (9.29) ... (9.31) виключають або всі c_i, d_i (СЛАР відносно b_i), або всі b_i, d_i (СЛАР відносно c_i). Розглянемо тільки другий варіант.

Щоб виключити всі b_i, d_i, спочатку (9.28) ділять на $h_{i+1}/6$, а (9.29) – на $h_i/6$. Потім з першого отриманого виразу віднімають другий:

$$6\left(\frac{y_{i+1} - y_i}{h_{i+1}} - \frac{y_i - y_{i-1}}{h_i}\right) = 6(b_{i+1} - b_i) - 3(h_{i+1}c_{i+1} - h_i c_i) + h_{i+1}^2 d_{i+1} - h_i^2 d_i. \tag{9.32}$$

З рівняння (9.30) $b_{i+1} - b_i = h_{i+1}c_{i+1} - h_{i+1}^2 d_{i+1}/2$, а з (9.31) $d_i = (c_i - c_{i-1})/h_i$ та $d_{i+1} = (c_{i+1} - c_i)/h_{i+1}$. Якщо все це підставити у (9.32), результат поділити на $(h_i + h_{i+1})$ та позначити

$$\mu_i = h_i/(h_i + h_{i+1}); \quad \lambda_i = h_{i+1}/(h_i + h_{i+1}) = 1 - \mu_i, \tag{9.33}$$

то остаточно утворюється рівняння відносно c_i:

$$\mu_i c_{i-1} + 2c_i + \lambda_i c_{i+1} = \frac{6}{h_i + h_{i+1}}\left(\frac{y_{i+1} - y_i}{h_{i+1}} - \frac{y_i - y_{i-1}}{h_i}\right); \quad i = 1,2,...,N-1. \tag{9.34}$$

Коли всі c_i будуть знайдені, то інші коефіцієнти можна підрахувати з формул, які виводяться з (9.31) та (9.30):

$$d_i = \frac{c_i - c_{i-1}}{h_i}; \quad b_i = \frac{h_i c_i}{2} - \frac{h_i^2 d_i}{6} + \frac{y_i - y_{i-1}}{h_i}; \quad i = 1,2,...,N. \tag{9.35}$$

Послідовно одержимо СЛАР для всіх вказаних граничних умов.

У випадку призначення І-го варіанта ГУ потрібно застосувати фіксовану величину $s'(\alpha)$. Щоб її залучити, вводять фіктивний нульовий сегмент (тобто при $i=0$), довжина якого $h_0 \to 0$. Дійсно, з другого виразу (9.35) при $i=0$ та $h_0 \to 0$ випливає, що $b_0 = \lim\limits_{h_0 \to 0}(y_0 - y_{-1})/h_0$. За смислом це відповідає визначенню першої похідної. Раніше отримали, що $b_i = s_i'(z_i) = y_i'$. Отже, введення фіктивного сегмента з $i=0$ та $h_0 \to 0$ дозволяє залучити для призначення ГУ ті з розглянутих вище рівнянь, які потрібні.

З (9.33) при $i=0$ та $h_0 \to 0$ випливає, що $\mu_0 = 0$, а $\lambda_1 = 1$. Тому з (9.34) при $i=0$ отримують, що

$$2c_0 + c_1 = \frac{6}{h_1}\left(\frac{y_1 - y_0}{h_1} - y'_\alpha\right). \tag{9.36}$$

Ще одне рівняння отримують з (9.35) при $i=N$, підставляючи перший вираз у другий та приймаючи, згідно з ГУ варіанта І $s'(\beta) = f'(\beta) = b_N = y'_\beta$. Після приведення подібних та множення на $6/h_N$ остаточно:

$$c_{N-1} + 2c_N = \frac{6}{h_N}\left(y'_\beta - \frac{y_N - y_{N-1}}{h_N}\right). \tag{9.37}$$

Отже, формули (9.36), (9.34) та (9.37) створюють СЛАР із $N+1$ рівнянь відносно c_i, $i=0,1,...,N$ у випадку І-го варіанта ГУ.

У випадку призначення ІІ-го варіанта ГУ все значно простіше, оскільки $s''(\alpha) = c_0 = y''_\alpha$; $s''(\beta) = c_N = y''_\beta$. Згідно з (9.34) відразу можемо записати СЛАР із $N-1$ рівнянь відносно c_i, $i=1,...,N-1$:

$$\begin{cases} 2c_1 + \lambda_1 c_2 = \dfrac{6}{h_1+h_2}\left(\dfrac{y_2 - y_1}{h_2} - \dfrac{y_1 - y_0}{h_1}\right) - \mu_1 y''_\alpha; \\ \mu_i c_{i-1} + 2c_i + \lambda_i c_{i+1} = \dfrac{6}{h_i + h_{i+1}}\left(\dfrac{y_{i+1} - y_i}{h_{i+1}} - \dfrac{y_i - y_{i-1}}{h_i}\right); \quad i = 2,...,N-2; \\ \mu_{N-1} c_{N-2} + 2c_{N-1} = \dfrac{6}{h_{N-1}+h_N}\left(\dfrac{y_N - y_{N-1}}{h_N} - \dfrac{y_{N-1} - y_{N-2}}{h_{N-1}}\right) - \lambda_{N-1} y''_N. \end{cases} \tag{9.38}$$

У випадку призначення ІІІ-го варіанта ГУ (періодичних) прирівнюються $s(\alpha) = a_0 = s(\beta) = a_N$, $s'(\alpha) = b_0 = s'(\beta) = b_N$ та $s''(\alpha) = c_0 = s''(\beta) = c_N$. У рівнянні (9.34) при $i=1$ замість c_0 застосовують c_N. Також застосовують рівняння (9.34) при $i=N$, причому замість c_{N+1} застосовують c_1 (з умов періодичності всі значення у вузлах $N+1$, $N+2$, ... дорівнюють відповідним значенням у вузлах з номерами, на N меншими, тобто у вузлах $1,2,...$). Остаточно одержують СЛАР із N рівнянь відносно c_i, $i=1,...,N$:

$$\begin{cases} 2c_1+\lambda_1 c_2+\mu_1 c_N=\dfrac{6}{h_1+h_2}\left(\dfrac{y_2-y_1}{h_2}-\dfrac{y_1-y_0}{h_1}\right); \\ \mu_i c_{i-1}+2c_i+\lambda_i c_{i+1}=\dfrac{6}{h_i+h_{i+1}}\left(\dfrac{y_{i+1}-y_i}{h_{i+1}}-\dfrac{y_i-y_{i-1}}{h_i}\right); \quad i=2,...,N-1; \\ \lambda_N c_1+\mu_N c_{N-1}+2c_N=\dfrac{6}{h_{N-1}+h_N}\left(\dfrac{y_N-y_{N-1}}{h_N}-\dfrac{y_{N-1}-y_{N-2}}{h_{N-1}}\right). \end{cases} \tag{9.39}$$

У випадку призначення IV-го варіанта ГУ маємо $d_m=d_{m+1}$, тому з першого виразу (9.35) із застосуванням (9.33) можна отримати, що

$$c_m=\lambda_m c_{m-1}+\mu_m c_{m+1}. \tag{9.40}$$

З рівняння (9.34) потрібно виключити c_0 та c_N. Із (9.40):

$$c_0=(c_1-\mu_1 c_2)/\lambda_1; \quad c_N=(c_{N-1}-\lambda_{N-1}c_{N-2})/\mu_{N-1}. \tag{9.41}$$

З другого рівняння (9.33) $\mu_i+\lambda_i=1$, тому

$$2\lambda_i+\mu_i=1+\lambda_i; \quad \lambda_i^2-\mu_i^2=(\lambda_i-\mu_i)(\lambda_i+\mu_i)=(\lambda_i-\mu_i). \tag{9.42}$$

Підставляючи перший вираз з (9.41) у (9.34) при $i=1$, після проведення множення його на λ_1 та застосування рівнянь із (9.42), остаточно мають, що

$$(1+\lambda_1)c_1+(\lambda_1-\mu_1)c_2=\frac{6\lambda_1}{h_1+h_2}\left(\frac{y_2-y_1}{h_2}-\frac{y_1-y_0}{h_1}\right). \tag{9.43}$$

Підставляючи другий вираз з (9.41) у (9.34) при $i=N-1$, після множення його на μ_{N-1} та застосування рівнянь з (9.42), остаточно мають, що

$$(\mu_{N-1}-\lambda_{N-1})c_{N-2}+(1+\mu_{N-1})c_{N-1}=\frac{6\mu_{N-1}}{h_{N-1}+h_N}\left(\frac{y_N-y_{N-1}}{h_N}-\frac{y_{N-1}-y_{N-2}}{h_{N-1}}\right). \tag{9.44}$$

Отже, формули (9.43), (9.34) при $i=2,3,...,N-2$ та (9.44) створюють СЛАР із $N-1$ рівнянь відносно c_i, $i=1,2,...,N-1$ у випадку IV-го варіанта ГУ, після отримання яких застосовуються формули (9.41).

Отримані СЛАР, а також вираз (9.26), часто записують інакше (не розглядаємо). Це робиться для зменшення кількості математичних операцій.

Оскільки матриця СЛАР має "діагональну перевагу" (алгебраїчна сума недіагональних коефіцієнтів будь-якого рядка матриці СЛАР менше відповідного діагонального коефіцієнта), то вона має лише один розв'язок. До того ж матриця СЛАР – трьохдіагональна, тому для отримання розв'язку зручно використовувати метод (схему) прогонки (див. підрозділ 5.4).

Доведено, що якщо функція $f(z)$ має четверту похідну, то точність інтерполяції сплайном функції та її похідних має таку оцінку:

$$\| f^{(k)}(z)-s^{(k)}(z)\| \le m_4\cdot h^{4-k}=O(h^{4-k}); \quad k=0,1,2,3, \tag{9.45}$$

де $m_4=\| f^{(IV)}(\xi)\|$ при деякому $\xi\in[\alpha,\beta]$, а $h=\max h_i$. Тобто кубічні сплайни $s(z)$ збігаються до функції $f(z)$ при необмеженому зростанні кількості вузлів N.

Деякі практичні рекомендації:

- доведено, що сплайн має різко виражені локальні властивості: якість наближення визначається в основному диференційними якостями функції $f(z)$, яку наближують, у деякому малому околі i-го вузла. Тому при локальних розривах похідних від $f(z)$ рекомендують точку розриву включати до вузлів апроксимації, а також по обидві її сторони вводити дуже короткі (десь у один відсоток від подальших) сегменти. Крім того, якщо розривна $f'(z)$, то потрібно у вузлі розриву призначити $d_i = 0$;
- якщо відносно функції $f(z)$, яку наближують, немає іншої інформації, ніж її вузлові значення, то потрібно застосовувати ГУ варіанта IV, а не II зі значеннями $s''(\alpha) = f''(\alpha) = 0$ та $s''(\beta) = f''(\beta) = 0$, як це іноді рекомендують, оскільки у останньому випадку на крайніх сегментах фактично получають перший порядок наближення;
- при застосуванні IV-го варіанта ГУ існує додаткова умова: перший та останній сегмент не повинні бути більшими, ніж другий та передостанній відповідно.

З викладеного очевидно, що основний недолік наближення сплайнами – це відсутність єдиного виразу СЛАР для всіх варіантів ГУ, тобто необхідність самостійно отримувати вираз СЛАР, якщо ГУ не співпадають з відомими варіантами. Інші недоліки: необхідно розв'язувати СЛАР, для цього мати відповідне програмне забезпечення; на кожному сегменті діапазону $[\alpha, \beta]$ потрібно застосовувати свої коефіцієнти, які для цього потрібно зберігати. Але ці недоліки компенсуються відмінною якістю наближення. Можливість залучення граничних умов тільки покращує ситуацію.

Примітка 9.5. При застосуванні іншої форми представлення сплайнів – *B*-сплайнів, потрібно менше запам'ятовувати даних, але загальна кількість дій збільшується. Крім того, форма *B*-сплайнів незручна у випадку IV-го варіанта ГУ, не гарантує добру обумовленість матриці СЛАР у загальному випадку, коли сегменти h_i значно відрізняються.

9.7. Найкраще наближення

Проблемою найкращого наближення називають задачу наближення функції $f(z)$ у діапазоні значень $z \in [\alpha, \beta]$ за допомогою системи лінійно незалежних функцій $\varphi_n(z)$; $n = 1,...,N$:

$$f(z) \approx \tilde{f}(z) = \sum_{n=1}^{N} c_n \cdot \varphi_n(z) \qquad (9.46\text{-а})$$

або

$$f(z) \approx \tilde{f}(z) = \psi(z) + \sum_{n=1}^{N} c_n \cdot \varphi_n(z), \qquad (9.46\text{-б})$$

які відповідають властивостям повноти.

Примітка 9.6. Оскільки кількість елементів N не є заздалегідь фіксованою величиною, завжди можна почати нумерацію елементів з нуля, зафіксувати c_0, позначити $\psi(z) \equiv c_0 \cdot \varphi_0(z)$ та винести $\psi(z)$ із під знаку додавання. Завжди можна легко призначити функцію $\psi(z)$ нульовою або такою, що задовольняє граничним умовам $\psi(\alpha) = f(\alpha)$ та $\psi(\beta) = f(\beta)$. Наприклад, лінійною: $\psi(z) = \lambda + \mu \cdot z$, де $\lambda = [f(\alpha) \cdot \beta - f(\beta) \cdot \alpha)]/(\beta - \alpha)$ та $\mu = [f(\beta) - f(\alpha)]/(\beta - \alpha)$. Але потрібно мати на увазі, що при використанні функції $\psi(z)$ для задоволення граничних умов потрібно обирати всі $\varphi_n(z)$ такими, що всі $\varphi_n(\alpha) = \varphi_n(\beta) = 0$, інакше на границях діапазону прийдеться призначати всі $c_n = 0$ (це випливає із умови лінійної незалежності функцій $\varphi_n(z)$, див. підрозділ 2.1). Оскільки c_n єдині на всьому діапазоні, то це фактично буде означати, що додаткові елементи не потрібні, а функція $\psi(z)$ і є найкращим наближенням функції $f(z)$. Надалі ми будемо використовувати більш загальний вираз (9.46-б), але на практиці частіше використовують варіант (9.46-а), тобто (9.46-б) при $\psi(z) \equiv 0$.

Коефіцієнти c_n потрібно знайти за умови, що

$$\| r(z) \| \to \min, \tag{9.47}$$

де позначена похибка наближення

$$r(z) = f(z) - \tilde{f}(z) = f(z) - \psi(z) - \sum_{n=1}^{N} c_n \cdot \varphi_n(z); \tag{9.48}$$

тобто норма похибки прямує до нуля при $N \to \infty$ як наслідок повноти.

Елементи $\varphi_n(z)$ називають *базисними* (іноді – пробними) функціями.

Доведено, що в гільбертовому просторі H таке найкраще наближення $\tilde{f}(z)$ існує та є одиничним. У залежності від того, який варіант гільбертового простору буде обрано (як задано скалярний добуток та норму у просторі), можна формулювати різні варіанти знаходження коефіцієнтів c_n з умови (9.47). Характерним прикладом такого простору є простір $\Omega_2(\alpha, \beta) \subset H$ дійсних функцій $f(z)$, які інтегруються з квадратом на $[\alpha, \beta]$, причому скалярний добуток та норма

$$(r, s) = \int_{\alpha}^{\beta} r(z) s(z) dz; \quad \| r \| = \sqrt{(r, r)} = \sqrt{\int_{\alpha}^{\beta} r^2(z) dz}. \tag{9.49}$$

Розглянемо основні методи побудови найкращого наближення, що використовують варіант (9.49).

9.7.1. Метод найменших квадратів

Метод практично одночасно був розроблений німецьким вченим Гауссом (K.F. Gauss) та французьким вченим Лежандром (A.M. Legendre) на самому початку XIX сторіччя. У сучасному розумінні він вимагає мінімізувати інтеграл від квадрату похибки по всій області визначення:

$$F = \int_{\alpha}^{\beta} r^2(z) dz. \tag{9.50}$$

Існує *основна лема фізики суцільних середовищ*: якщо у цілком щільній області Ω для будь-якого $\Omega_i \subset \Omega$ інтеграл $\int\limits_{\Omega_i} \theta d\Omega = 0$, то функція $\theta \equiv 0$ у всій Ω.

У виразі (9.50) в якості функцій під інтегралом виступає квадрат похибки наближення. При точному наближенні похибка наближення дорівнює нулю на будь-якому відрізку діапазону визначення функції, яка наближується. Фактично це зазвичай не відбувається, але той факт, що функція під інтегралом є позитивною, дозволяє використовувати мінімум функціонала (9.50) як умову найкращого наближення.

Умова мінімізації набуває вигляд $\partial F / \partial c_m = 0$; $m = 1,2,...,N$. Оскільки $\partial r / \partial c_m = \varphi_m(z)$, то легко отримати, що цей мінімум досягається, якщо

$$\int\limits_{\alpha}^{\beta} r(z) \cdot \varphi_m(z) dz = \int\limits_{\alpha}^{\beta} \left[f(z) - \psi(z) - \sum_{n=1}^{N} c_n \cdot \varphi_n(z) \right] \cdot \varphi_m(z) dz = 0; \quad m = 1,...,N. \qquad (9.51)$$

Вираз (9.51) є системою лінійних алгебраїчних рівнянь (СЛАР) вигляду

$$A_{mn} c_n = b_m; \quad m,n = 1,...,N, \qquad (9.52)$$

де компоненти матриці та вектора правої частини обчислюються як

$$A_{nm} = A_{mn} = \int\limits_{\alpha}^{\beta} \varphi_n(z) \cdot \varphi_m(z) dz; \quad b_m = \int\limits_{\alpha}^{\beta} \left[f(z) - \psi(z) \right] \cdot \varphi_m(z) dz; \quad m,n = 1,...,N. \qquad (9.53)$$

Матриця A_{mn} для системи лінійно незалежних базисних функцій $\varphi_n(z)$, $n = 1,...,N$ є симетричною та позитивно визначеною, зветься *матрицею Грама*.

Примітка 9.7. Якщо функція, що наближується, є табличною, то в формулах (9.49) ... (9.51) і (9.53) інтеграли замінюються на суми $\sum\limits_{k=1}^{M}$, причому замість безперервної змінної z застосовується дискретна z_k, $k = 1,...,M$, де кількість точок у таблиці, причому потрібно, щоб було $M \geq N$.

Примітка 9.8. При $M = N$ задача вироджується у задачу інтерполювання, в якій обов'язковою вимогою є $\tilde{f}(z_k) = f(z_k)$.

Примітка 9.9. Якщо $M > N$, то відбувається *згладжування* функції, яка наближується: локальні викиди ("піки" або "провали") зменшуються. Це відбувається тим сильніше, чим більше вузлів знаходиться у ближньому околі локального викиду, а також чим сильніше відрізняється M від N. Саме для випадку $M > N$ й був розроблений цей метод (обробка експериментальних даних).

Як окремий, але дуже вдалий випадок базисних функцій, є тригонометричні функції. Якщо провести заміну $\xi = z + \alpha$, то можна обрати:

$$\varphi_n = \sin(n\pi\xi / L), \qquad (9.54)$$

де $L = \beta - \alpha$ – довжина області в напрямку ξ (та z).

Внаслідок ортогональності функцій (9.54), яка виражається у тому, що

$$\int\limits_{0}^{L} \sin(m\pi\xi / L) \cdot \sin(n\pi\xi / L) d\xi = \begin{cases} L/2, & m = n; \\ 0, & m \neq n, \end{cases} \qquad (9.55)$$

матриця СЛАР утворюється діагональною ($L/2$ на діагоналі), тому відразу ж можна отримати шукані коефіцієнти

$$c_n = \frac{2}{L}\int_0^L \left[f(\xi) - \psi(\xi)\right] \cdot \sin(n\pi\xi / L) d\xi . \qquad (9.56)$$

Такі коефіцієнти c_n називають коефіцієнтами Фур'є ортогональної системи базисних функцій (9.54), а розклад (9.46) з ортогональними базисними функціями – многочленом Фур'є.

Таким чином, ортогональність базисних функцій є важливою властивістю, яка дозволяє різко прискорити отримання розв'язку СЛАР.

Приклад 9.7. Маємо таблицю $f_n(x_n)$, $n = 1,..., N > 2$, отриману в експерименті. Потрібно її апроксимувати лінійною функцією $f(x) = A + Bx$.

Застосуємо метод найменших квадратів.

В кожній точці маємо деяку похибку $R_n = A + Bx_n - f_n$. Згідно з методом, складаємо функціонал $F = \sum_{n=1}^{N} R_n^2 = \sum_{n=1}^{N} (A + Bx_n - f_n)^2 \to \min$. Умови його мінімізації $\partial F / \partial A = 0$ та $\partial F / \partial B = 0$. Тому

$$\partial F / \partial A = \partial \sum_{n=1}^{N} (A + Bx_n - f_n)^2 / \partial A = 2\sum_{n=1}^{N} (A + Bx_n - f_n) = 0 ;$$

$$\partial F / \partial B = \partial \sum_{n=1}^{N} (A + Bx_n - f_n)^2 / \partial B = 2\sum_{n=1}^{N} ((A + Bx_n - f_n) \cdot x_n) = 0 .$$

Це є система з двох алгебраїчних рівнянь відносно невідомих A та B:

$$\begin{bmatrix} N & \sum_{n=1}^{N} x_n \\ \sum_{n=1}^{N} x_n & \sum_{n=1}^{N} x_n^2 \end{bmatrix} \begin{pmatrix} A \\ B \end{pmatrix} = \begin{pmatrix} \sum_{n=1}^{N} f_n \\ \sum_{n=1}^{N} (f_n x_n) \end{pmatrix}, \quad \text{або для скорочення} \quad \begin{bmatrix} a_{11} & a_{12} \\ a_{12} & a_{22} \end{bmatrix} \begin{pmatrix} A \\ B \end{pmatrix} = \begin{pmatrix} b_1 \\ b_2 \end{pmatrix}.$$

За схемами Крамера або Гаусса отримаємо, що

$B = (a_{11}b_2 - a_{12}b_1) / (a_{11}a_{22} - a_{12}^2)$; $\quad A = (b_1 - a_{12}B) / a_{11}$.

Щоб побачити графік функції $f(x) = A + Bx$, достатньо обрати дві точки, наприклад, $x = 0$ та $x = \xi$, де $\xi \geq \max\{x_n\}$, обчислити $f(0) = A$ та $f(\xi) = A + B\xi$ проставити ці значення на графіку та провести через них пряму лінію. Це буде найкраще наближення табличних даних $f_n(x_n)$, $n = 1,..., N > 2$ лінійною функцією.

9.7.2. Метод Релея

Розглянемо в гільбертовому просторі H квадрат норми (9.47) з урахуванням (9.48) та позначення $\widehat{f} = f - \psi$:

$$\|r\|_H^2 = \left\| \widehat{f} - \sum_{n=1}^{N} c_n \cdot \varphi_n \right\|_H^2 = \| \widehat{f} \|_H^2 - 2\sum_{n=1}^{N} c_n \widehat{f} \varphi_n + \sum_{m=1}^{N} c_m \varphi_m \sum_{n=1}^{N} c_n \varphi_n \to \min . \qquad (9.57)$$

Позначимо:

$$b_m = (\widehat{f}, \varphi_m)_H ; \quad a_{mn} = (\varphi_m, \varphi_n)_H ; \quad \vec{c} = \{c_1, c_2, ..., c_N\}^T ; \quad \vec{b} = \{b_1, b_2, ..., b_N\}^T . \qquad (9.58)$$

Тоді вираз (9.57) можна записати у вигляді

$$\|r\|_H^2 = \|\hat{f}\|_H^2 - 2(\vec{c},\vec{b}) + (A\vec{c},\vec{c}) \to \min. \qquad (9.59)$$

Оскільки $\|\hat{f}\|_H^2$ не можна змінювати, то функціонал (9.59) може отримати мінімальне значення лише за умови, що

$$(A\vec{c},\vec{c}) - 2(\vec{c},\vec{b}) \to \min \quad \text{або} \quad F(\vec{c}) = \frac{1}{2}(A\vec{c},\vec{c}) - (\vec{c},\vec{b}) \to \min. \qquad (9.60)$$

Оператор A є позитивно визначеною симетричною матрицею Грама з елементами $a_{mn} = (\varphi_m, \varphi_n)_H$; $m,n = 1,...,N$, тобто ще й лінійним, тому повністю задовольняє умовам Теореми 2 підрозділу 2.4. Згідно з цією теоремою, функціонал $F(\vec{c})$ (див. формули (9.60) і (2.24)) мінімізується вектором $\vec{c}$.

У випадку простору $\Omega_2(\alpha,\beta) \subset H$ дійсних функцій $f(z)$, які інтегруються з квадратом на $[\alpha,\beta]$, отримаємо компоненти СЛАР (9.52) з компонентами (9.53) при наближенні функції $f(z)$ на $[\alpha,\beta]$.

Фактично цей метод був створений Релеєм (Sir Jon William Strutt, Lord Rayleigh) у 1877 році. Теоретично він був обґрунтований у 1918 році М.М. Криловим (умови збіжності).

Нехай $\lambda = \min F(\vec{c})$. Послідовність (9.46) називають такою, що мінімізує, якщо $\lim\limits_{N\to\infty} F(\vec{c}) = \lambda$. Доведено, що послідовність (9.57) дійсно є такою, що мінімізує цей функціонал.

9.7.3. Метод зважених похибок наближення

Метод створено Г.І. Петровим у 1938 ... 1940 рр. Автор виходив з фундаментальної теореми про проєкції (див. підрозділ 2.1), згідно з якою у випадку замкненого простору $\Omega_2(\alpha,\beta) \subset H$ дійсних функцій $f(z)$, які інтегруються з квадратом на $[\alpha,\beta]$, із застосуванням (9.48) умову (2.11) запишемо у вигляді

$$\|f(z) - \tilde{f}(z)\| < \|f(z) - f^{\#}(z)\|, \qquad (9.61)$$

де $f^{\#}(z)$ – будь-яка інша функція (не $\tilde{f}(z)$). Необхідною та достатньою умовою виконання цієї нерівності є ортогональність $r(z) = f(z) - \tilde{f}(z)$ будь-якому вектору $\vec{w} = \{w_1, w_2, ..., w_N\}_H^T \in \Omega_2$, тобто потрібно, щоб усі функціонали

$$F_m = \int_\alpha^\beta r(z)\cdot w_m(z)dz = 0; \quad m = 1,...,N. \qquad (9.62)$$

З урахуванням виразу (9.48) для $r(z)$ отримаємо СЛАР (9.52) з компонентами

$$A_{mn} = \int_\alpha^\beta \varphi_n(z)\cdot w_m(z)dz; \quad b_m = \int_\alpha^\beta [f(z) - \psi(z)]\cdot w_m(z)dz; \quad m,n = 1,...,N. \qquad (9.63)$$

Цей метод надає велику свободу для обирання компонент *вагового* вектора $\vec{w}$, тобто системи $w_m = w_m(z)$. Обов'язкова умова: це повинна бути повна за енергією лінійно незалежна система. Тому метод найменших квадратів (див. п. 9.7.1), в якому фактично в ролі компонент вагового вектора використовуються саме базисні функції, можна розглядати як окремий випадок методу зважених похибок наближення (МЗПН).

Компонентами вагового вектора також можуть бути дельта-функції Дірака (метод *поточкових колокацій*):

$$w_m = \delta(z - z_m) \tag{9.64}$$

із властивостями

$$\delta(z - z_m) = 0, \;\; z \neq z_m; \qquad \delta(z - z_m) = \infty, \;\; z = z_m; \qquad \int\limits_{z<z_m}^{z>z_m} \varphi(z)\delta(z - z_m)dz = \varphi(z_m). \tag{9.65}$$

Це еквівалентне тому, що похибка наближення вважається рівною нулю в заданих точках z_m. З урахуванням (9.64) і (9.65) з (9.63) отримаємо

$$A_{mn} = \varphi_n(z_m); \quad b_m = f(z_m) - \psi(z_m); \quad m, n = 1,..., N. \tag{9.66}$$

Ще один варіант – метод *колокацій по підобластям*. Вагові функції

$$w_m = \begin{cases} 1, & z_m < z < z_{m+1}; \\ 0, & z < z_m, \;\; z > z_{m+1}, \end{cases} \tag{9.67}$$

тому рівняння (9.62) відповідають деяким підобластям основної області. З (9.63):

$$A_{mn} = \int\limits_{z_m}^{z_{m+1}} \varphi_n(z)dz; \quad b_m = \int\limits_{z_m}^{z_{m+1}} \left[f(z) - \psi(z)\right]dz; \quad m, n = 1,..., N. \tag{9.68}$$

І останній варіант обирання компонент вагових функцій, який розглянемо, це метод *Бубнова* (І.Г. Бубнов, 1913 р.), коли в ролі вагових функцій використовуються саме базисні функції:

$$w_m(z) = \varphi_m(z). \tag{9.69}$$

З (9.63), з використанням (9.69), випливають рівняння (9.53). Матриця СЛАР є симетричною, що полегшує її розв'язування.

Отже, МЗПН фактично є узагальненням багатьох методів найкращого наближення функцій.

9.8. Інші варіанти інтерполювання або наближення функцій

9.8.1. Інтерполювання многочленом Ерміта

У випадку, коли крім значень $y_i = f(z_i)$ при інтерполюванні потрібно врахувати відомі похідні функції у вузлах, тобто $y_i^{(j)} = f^{(j)}(z_i)$, застосовують інтерполяційний многочлен Ерміта.

Отже, в кожному вузлі $z_i \in [\alpha, \beta]$, $i = 0,1,..., N$ задані

$$y_i^{(j)} = f^{(j)}(z_i); \quad j = 0,1,...M_i - 1, \tag{9.70}$$

де $y_i^{(0)} = f^{(0)}(z_i) = y_i = f(z_i)$, а число M_i є порядком похідної у вузлі з номером i, яка вже відома, має назву *кратності* i–го вузла. Тобто відомо $m = M_0 + M_1 + ... + M_N$ значень функцій та їхніх похідних у вузлах. Многочлен, який задовольняє цим вимогам, є інтерполяційним многочленом Ерміта:

$$H_n(z) = \sum_{i=0}^{N} \sum_{j=0}^{M_i - 1} f^{(j)}(z_i) \cdot c_{ij}(z), \tag{9.71}$$

де $n = m - 1$, який фактично є лінійною комбінацією звичайних *степеневих поліномів* $c_{ij}(z)$ степені n, а також коефіцієнтів, заданих (9.70).

Відомо, що многочлени Ерміта є розв'язком диференційного рівняння $f''(z) - 2zf'(z) + 2nf(z) = 0$.

Похибка інтерполювання многочленом Ерміта:

$$f(z) - H_n(z) = \frac{f^{(n+1)}(\xi)}{(n+1)!}(z - z_0)^{M_0}(z - z_1)^{M_1} \cdot \ldots \cdot (z - z_M)^{M_N}; \quad \xi \in [\alpha, \beta]. \tag{9.72}$$

Навіть при невеликих значеннях N та M_j докладні вирази інтерполяційного многочлена Ерміта є складними, тому тут їх не наводимо.

9.8.2. Тригонометричне інтерполювання періодичної функції

Якщо функція $f(z)$, яку потрібно наблизити, є періодичною, то окрім сплайнової апроксимації з ГУ варіанта ІІІ (див. підрозділ 9.6) можна застосовувати тригонометричний многочлен. У випадку непарної кількості вузлів (нагадаємо, що $z_i \in [\alpha, \beta]$, причому $\alpha = z_0 < z_1 < \ldots < z_{N-1} < z_N = \beta$):

$$T_n(z) = \alpha_0 + \sum_{j=1}^{n}\left(a_j \cos(\pi jz / L) + b_j \sin(\pi jz / L)\right), \tag{9.73-а}$$

де $L = \beta - \alpha$, а число $n = N/2$. Якщо кількість вузлів є парною, то коефіцієнт α_0 відкидається:

$$T_n(z) = \sum_{j=1}^{n}\left(a_j \cos(\pi jz / L) + b_j \sin(\pi jz / L)\right), \tag{9.73-б}$$

причому число $n = (N+1)/2$.

Коефіцієнти a_j та b_j знаходяться із системи рівнянь

$$T_n(z_i) = f(z_i); \quad i = 0,1,\ldots,N. \tag{9.74}$$

9.8.3. Інтерполювання раціональними функціями

У цьому випадку функція $f(z)$ наближується виразом

$$\psi_{mn}(z) = \frac{a_m z^m + a_{m-1} z^{m-1} + \ldots + a_0}{z^n + b_{n-1} z^{n-1} + \ldots + b_0} = \frac{\sum_{k=0}^{m}\left(a_k \cdot z^k\right)}{z^n + \sum_{k=0}^{n-1}\left(b_k \cdot z^k\right)}, \tag{9.75}$$

де m та n є фіксованими числами. Коефіцієнти $a_0, \ldots, a_m, b_0, \ldots, b_{n-1}$ у кількості $m + n + 1$ знаходяться за умови

$$\psi_{mn}(z_i) = f(z_i); \quad i = 0,1,\ldots,N, \tag{9.76}$$

тобто із СЛАР з $N+1$ рівнянь. Щоб ця СЛАР мала квадратну матрицю, потрібно, щоб $m + n + 1 = N + 1$ або $m + n = N$. СЛАР буде мати компоненти

$$\sum_{k=0}^{m}\left(a_k \cdot (z_i)^k\right) - f(z_i) \cdot \sum_{k=0}^{n-1}\left(b_k \cdot (z_i)^k\right) = f(z_i) \cdot (z_i)^n; \quad i = 0,1,\ldots,m+n. \tag{9.77}$$

При застосуванні цього варіанта необхідно стежити, щоб знаменник виразу (9.75) ніде в $[\alpha,\beta]$ не вироджувався в нуль, а також щоб в $[\alpha,\beta]$ чисельник у (9.75) не ділився на знаменник без остачі (ситуація залежить від кількості та положення вузлів).

Якщо $m=n=1$, тобто при наявності трьох вузлів, інтерполяція виразом (9.75) називається дрібно-лінійною інтерполяцією. Саме у цьому випадку часто зустрічаються вказані вище вироджені ситуації.

9.9. Наближене диференціювання функцій, заданих таблицею

Проблема виникає, коли є функція, що задана таблицею, і необхідно обчислити значення похідної (диференціала) функції в заданій точці.

У підрозділі 9.2, використовуючи наближену формулу (9.4) для скінченних різниць різних порядків, отримали формули для перших чотирьох похідних, кожна з яких використовує мінімальну кількість заданих таблицею значень функції (стандартний шаблон).

Інакше це завдання можна розв'язати шляхом диференціювання полінома, що інтерполює функцію, задану таблицею. При цьому кількість вузлових значень, що використовуються, можна збільшити.

Нехай z – точка, в якій потрібно обчислити похідну. Оберемо у її найближчому околі точку z_0, яка є в таблиці. Позначимо $h=z_{i+1}-z_i$ – крок аргументу таблиці (тут він повинен бути постійним) та введемо новий аргумент $q=(z-z_0)/h$, що відповідає z. Визначимо, що $dq/dz=1/h$.

9.9.1. Формула наближеного диференціювання, що заснована на першій інтерполяційній формулі Ньютона

Використаємо першу інтерполяційну формулу Ньютона (9.8) і проведемо її диференціювання. Отримаємо, що:

$$y'(z)\approx\frac{1}{h}\left[\Delta y_0+\frac{2q-1}{2}\Delta^{(2)}y_0+\frac{3q^2-6q+2}{6}\Delta^{(3)}y_0+\frac{2q^3-9q^2+11q-3}{12}\Delta^{(4)}y_0+\ldots\right]; \quad (9.78)$$

$$y''(z)\approx\frac{1}{h}\left[\Delta^{(2)}y_0+(q-1)\Delta^{(3)}y_0+\frac{6q^3-18q+11}{12}\Delta^{(4)}y_0+\ldots\right] \quad (9.79)$$

з похибкою $R_n^{(1)}(z_0)\approx(-1)^n\Delta^{(n+1)}y_0/[h(n+1)]$.

При $z=z_0$ (у точках таблиці) $q=0$, тому:

$$y'(z_0)\approx\frac{1}{h}\left[\Delta y_0-\frac{1}{2}\Delta^{(2)}y_0+\frac{1}{3}\Delta^{(3)}y_0-\frac{1}{4}\Delta^{(4)}y_0+\ldots\right]; \quad (9.80)$$

$$y''(z_0)\approx\frac{1}{h}\left[\Delta^{(2)}y_0-\Delta^{(3)}y_0+\frac{11}{12}\Delta^{(4)}y_0+\ldots\right]. \quad (9.81)$$

Надійність обчислення кожної наступної похідної – менша, ніж попередніх. Тому немає особливого сенсу отримувати їх вирази та застосовувати.

Приклад 9.8. Візьмемо функцію $y = \log z$. Припустимо, що маємо значення цієї функції тільки в точках $z = 50, 55, 60, 65$. Потрібно обчислити похідну при $z = 50$.

Будуємо, наприклад, горизонтальну таблицю різниць:

z	$y = \log z$	$\Delta^{(1)} y$	$\Delta^{(2)} y$	$\Delta^{(3)} y$
50	1.6990	0.0414	-0.0036	-0.0005
55	1.7404	0.0378	-0.0031	-
60	1.7782	0.0347	-	-
65	1.8129	-	-	-

Визначимо, що $h = 5$, а $q = 0$. Тому, застосовуючи формулу (9.80), отримаємо, що $y'(50) \approx \frac{1}{5}\left[0.0414 - \frac{-0.0036}{2} + \frac{-0.0005}{3}\right] \approx 0.0087$. Більш точне значення обчислимо за відомою формулою $(\log z)' = 1/[z \cdot \ln(10)]$, отримаємо, що $(\log(50))' \approx 0.008686$. Тобто у отриманому чисельним інтегруванням значенні всі цифри є вірними.

9.9.2. Формула наближеного диференціювання, що заснована на інтерполяційній формулі Стірлінга

Введемо позначення:

$$\Delta\bar{y} = (\Delta y_{-1} + \Delta y_0)/2; \quad \Delta^{(3)}\bar{y} = (\Delta^{(3)} y_{-2} + \Delta^{(3)} y_{-1})/3; \quad \Delta^{(5)}\bar{y} = (\Delta^{(5)} y_{-3} + \Delta^{(5)} y_{-2})/5. \quad (9.82)$$

Формулу (9.14) перепишемо у вигляді:

$$P_n(z) = y_0 + q\Delta\bar{y} + \frac{q^2}{2!}\Delta^{(2)} y_{-1} + \frac{q(q^2-1^2)}{3!}\Delta^{(3)}\bar{y} + \frac{q^2(q^2-1)}{4!}\Delta^{(4)} y_{-2} + \frac{q(q^2-1^2)(q^2-2^2)}{5!}\Delta^{(5)}\bar{y} + \ldots. \quad (9.83)$$

Проведемо її диференціювання. Отримаємо, що:

$$y'(z) \approx \frac{1}{h}\left[\Delta\bar{y} + q\Delta^{(2)} y_{-1} + \frac{3q^2-1}{6}\Delta^{(3)}\bar{y} + \frac{2q^3-q}{12}\Delta^{(4)} y_{-2} + \ldots\right]; \quad (9.84)$$

$$y''(z) \approx \frac{1}{h}\left[\Delta^{(2)} y_{-1} + q\Delta^{(3)}\bar{y} + \frac{q^2-1}{2}\Delta^{(4)} y_{-2} + \ldots\right]. \quad (9.85)$$

Ці формули рекомендують застосовувати (як і формулу Стірлінга) при $|q| \le 0.5$. При $z = z_0$ (у точках таблиці) $q = 0$, тому:

$$y'(z) \approx \frac{1}{h}\left[\Delta\bar{y} - \frac{1}{6}\Delta^{(3)}\Delta\bar{y} + \frac{1}{30}\Delta^{(5)}\bar{y} - \ldots\right]; \quad (9.86)$$

$$y''(z) \approx \frac{1}{h}\left[\Delta^{(2)} y_{-1} - \frac{1}{12}\Delta^{(4)} y_{-2} + \frac{1}{90}\Delta^{(6)} y_{-3} - \ldots\right]. \quad (9.87)$$

Як і формула Стірлінга, ці формули є центрально-різницевими, тобто можуть застосовуватися на визначеній відстані від країв таблиці та у середині таблиці, оскільки потребують значень з обох сторін актуальної точки.

9.9.3. Диференціювання функції із застосуванням кубічних сплайнів

Диференціювання функції $f(z)$, інтерпольованої із застосуванням кубічних сплайнів, проводиться простим диференціюванням виразу сплайну. Зокрема, якщо застосувати кубічний сплайн класу $s(z) \in C^2[\alpha, \beta]$, тобто з дефектом $k = 1$ (9.26), тобто

$$s_i(z) = f(z_i) + b_i(z - z_i) + \frac{c_i}{2}(z - z_i)^2 + \frac{d_i}{6}(z - z_i)^3; \quad z \in [z_{i-1}, z_i]; \quad i = 1, 2..., N, \quad (9.88)$$

то легко одержати, що

$$s_i'(z) = b_i + c_i(z - z_i) + \frac{d_i}{2}(z - z_i)^2; \quad z \in [z_{i-1}, z_i]; \quad i = 1, 2..., N, \quad (9.89)$$

$$s_i''(z) = c_i + d_i(z - z_i); \quad z \in [z_{i-1}, z_i]; \quad i = 1, 2..., N, \quad (9.90)$$

$$s_i'''(z) = d_i; \quad z \in [z_{i-1}, z_i]; \quad i = 1, 2..., N. \quad (9.91)$$

У цих формулах коефіцієнти b_i та d_i утворюються із застосуванням формул (9.35), а c_i – як результат розв'язання СЛАР, що відповідає граничним умовам обраного варіанта (див. підрозділ 9.6).

9.9.4. Некоректність операції чисельного диференціювання

Зазвичай значення функції $f(z_i)$ в таблиці мають деякі похибки, викликані або наближеним обчисленням або усіканням кількості значущих цифр. Позначимо як Δ_i такі похибки у вузлах. Розглянемо простий випадок: обчислення першої похідної формулою $f'(z) \approx (y_{i+1} - y_i)/h$ (див. підрозділ 9.2). Максимальна можлива похибка від такого роду похибок буде такою:

$$|\overline{\Delta}_{f'}|_{\max} = \max\{[(y_{i+1} + \Delta_{i+1}) - (y_i - \Delta_i)]/h - (y_{i+1} - y_i)/h\} = \max\{(\Delta_{i+1} - \Delta_i)/h\} = 2\Delta/h, \quad (9.92)$$

де $\Delta = \max\{\Delta_{i+1}, \Delta_i\}$. Одночасно максимальна похибка апроксимації саме похідної $f'(z)$ має значення $\|\Delta_{f'}\| = hM_2/2$, де M_2 є максимальне значення другої похідної в діапазоні $z \in [z_{i+1}, z_i]$. Природно вимагати, щоб похибка $|\overline{\Delta}_{f'}|_{\max}$ не перевищувала похибки $|\Delta_{f'}|_{\max}$, тобто щоб $2\Delta/h \le hM_2/2$. Звідси можна записати вимоги або для Δ або для h, коли Δ не можна змінити:

$$\Delta \le M_2(h/2)^2 = O(h^2) \quad \text{або} \quad h \ge 2\sqrt{\Delta/M_2}, \quad (9.93)$$

причому $|\Delta_{f'}|_{\max} = O(h)$.

Аналогічні (9.93) обмеження можна отримати і для інших формул чисельного диференціювання. Загальним є той факт, що з підвищенням порядку похідної ситуація погіршується. Тому у загальному випадку операція чисельного диференціювання вважається некоректною.

Контрольні питання до підрозділу 9.1

1. Чим розрізняються поняття відновлення, інтерполювання, екстраполювання та наближення функцій?

Контрольні питання до підрозділу 9.2

1. Які формули застосовуються у методі скінченних різниць для апроксимації похідних другого, третього та четвертого порядків?
2. Які властивості має інтерполяційний поліном n-ої степені при $n = N$?

Контрольні питання до підрозділу 9.3

1. В яких межах застосовують інтерполяційні формули Ньютона?
2. Чи мають інтерполяційні формули Ньютона формули для оцінювання похибки інтерполяції?

Контрольні питання до підрозділу 9.4

1. Як створюється інтерполяційна формула Стірлінга?

Контрольні питання до підрозділу 9.5

1. Які інтерполяційні формули застосовуються у випадку різного кроку між опорними вузлами?
2. Що це таке: поділені різниці?

Контрольні питання до підрозділу 9.6

1. Які властивості має кубічний сплайн класу C^2 з дефектом $k = 1$?
2. Як будується кубічний сплайн $s(z)$?
3. Назвіть практичні рекомендації щодо використання сплайнів.

Контрольні питання до підрозділу 9.7

1. Що називають проблемою найкращого наближення?
2. Про що йдеться в основній лемі фізики суцільних середовищ?
3. Який суттєвий недолік має метод найменших квадратів?
4. Які обмеження має метод Релєя?
5. Чому метод зважених похибок наближення вважають універсальним?

Контрольні питання до підрозділу 9.8

1. В яких випадках для найкращого наближення застосовують інтерполяційний многочлен Ерміта?
2. Які особливості є при тригонометричному інтерполюванні періодичної функції?
3. Які особливості є при інтерполюванні раціональними функціями?

Контрольні питання до підрозділу 9.9

1. Чи можна використовувати різноманітні формули інтерполювання при чисельному диференціюванні?
2. Чому операції чисельного диференціювання вважають некоректними?

Розділ 10

НАБЛИЖЕНЕ ІНТЕГРУВАННЯ ФУНКЦІЙ

10.1. Загальні положення

Формула Ньютона-Лейбніца

$$\int_a^b f(x)dx = F(b) - F(a); \quad F'(x) = f(x), \tag{10.1}$$

де $F(x)$ – первісна, дозволяє точно обчислити значення визначеного інтеграла. Однак, цього досить часто не вдається зробити, оскільки:

- не знайдено аналітичного виразу для первісної $F(x)$;
- функція $f(x)$ задана таблицею (не має аналітичного запису).

Тоді застосовують наближене інтегрування функцій. У загальному випадку процес побудови формул для наближеного інтегрування містить такі етапи:

а/ вихідна функція $f(x)$ замінюється на добуток двох інших, який наближує функцію, тобто приймається, що $f(x) \approx p(x) \cdot \varphi(x)$. При цьому *вагова* функція $p(x)$ повинна мати такі же особливості, як і функція $f(x)$, наприклад, мати таку ж швидкість збіжності до деякого значення. Якщо таких особливостей немає, то приймають $p(x) = 1$. Функція $\varphi(x)$ повинна бути *гладкою* функцією, яка добре наближується многочленами або раціональними функціями.

Тоді результат інтегрування можна записати у вигляді *квадратурної суми*

$$\int_a^b f(x)dx \approx \int_a^b p(x) \cdot \varphi(x)dx \approx \sum_{i=0}^{n} \alpha_i \cdot \psi(x_i), \tag{10.2}$$

де α_i – квадратурні ("вагові") коефіцієнти, $x_i \in [a,b]$ – вузли квадратури (точки інтегрування), а у якості функції $\psi(x)$ може виступати $\varphi(x)$ або інша функція. Зазвичай $\varphi(x)$ обирається у вигляді деякого полінома: Лагранжа, Лежандра, Чебишева тощо. При цьому й $\psi(x)$ – теж поліном. На момент інтегрування вагова функція $p(x)$ вважається фіксованою (визначеною), тому її вплив відображається у квадратурних коефіцієнтах α_i.

Отже, у загальному випадку може бути дві похибки. Перша пов'язана із заміною $f(x)$ на $p(x) \cdot \varphi(x)$ і може визначатися як

$$\int_a^b | f(x) - p(x) \cdot \varphi(x) | dx = \varepsilon(x) \le \Delta; \quad \Delta \ge 0. \tag{10.3}$$

Друга похибка пов'язана із заміною інтеграла сумою. Її визначають як

$$\int_a^b p(x) \cdot \varphi(x)dx - \sum_{i=0}^{n} \alpha_i \cdot \psi(x_i) = R_n(x). \tag{10.4}$$

Першу похибку не можна заздалегідь формалізувати, оскільки вона залежить від якості інтерполяції (див. Розділ 9. **Увага**: для інтерполяційних

функцій обов'язково потрібно виконувати умову $f(x_i) = p(x_i) \cdot \varphi(x_i)$). Тому при створенні формул наближеного інтегрування із застосуванням квадратурних сум зазвичай всю увагу зосереджують на визначенні другої похибки, тобто $R_n(x)$. Однак це не означає, що похибка (10.3) не має великого значення, оскільки точність розрахунків визначається сумарною похибкою.

б/ вибирають принципи, за якими призначаються $p(x) \cdot \varphi(x)$, n, α_i та x_i. Число n визначає точність формули, тому вважається фіксованим. Зазвичай зі збільшенням n точність зростає, але не завжди: в деяких формулах при великих значеннях n квадратурні коефіцієнти α_i мають різні знаки, що призводить до значного зниження точності. Що до вибору α_i та x_i, є три варіанти:

- попередньо визначеними є квадратурні ("вагові") коефіцієнти α_i, причому є такі способи їх призначення:
  - усі α_i є однаковими;
  - α_i визначаються з умови мінімізації похибок наближення (тоді повинні бути всі $\alpha_i > 0$);
  - α_i – симетричні в діапазоні інтегрування;
- попередньо визначеними є всі квадратурні вузли x_i (на рівномірних або нерівномірних сітках; зазвичай – на симетрично розташованих рівномірних).
- попередньо визначеними є деякі (не всі) квадратурні вузли x_i, причому є сенс обов'язково визначати вузли на границях області інтегрування.

Якщо хоча б одна з границь інтервалу не містить квадратурного вузла, то така квадратурна формула називається відкритою, інакше – закритою.

Як з'ясувалося, всі формули наближеного інтегрування, побудовані на *парній* кількості квадратурних вузлів, *симетричних* відносно центра інтервалу (симетричні формули), мають дещо кращу точність, ніж це випливає із загальних оцінок.

в/ для підвищення точності інтервал $[a,b]$ розбивають на m відрізків з кроком $h = (b-a)/m$, тоді

$$\int_a^b f(x)dx \approx \sum_{j=1}^{m} \int_{x_j}^{x_j+h} p_j(x) \cdot \varphi_j(x)dx . \qquad (10.5)$$

Квадратурними називають формули чисельного інтегрування *одинарних* інтегралів. Якщо інтеграл – подвійний або потрійний, то формули наближеного інтегрування функцій називаються *кубатурними*. Як і аналітичне інтегрування, їх наближене інтегрування проводиться послідовним подвійним або потрійним застосуванням квадратурних формул.

10.2. Поліноміальне інтегрування

У формулі (10.2) приймемо $p(x) \equiv 1$, оберемо $f(x) \approx \varphi(x) = x^m$; $m = 0,1,...,n$. Зажадаємо, щоб квадратурна сума була точною для всіх $m = 0,1,...,n$. Для прикладу отримаємо формулу поліноміального інтегрування у випадку, коли

$n=2$; для *симетрично* розташованих x_i в інтервалі $[a,b]=[0,1]$, причому оберемо $x_0=1/4$, $x_1=1/2$ та $x_2=3/4$, а $\psi(x)=\varphi(x)$. Спочатку проведемо аналітичне інтегрування, щоб отримати систему рівнянь. Результати обчислень наведені в таблиці 10.1.

Таблиця 10.1. Поліноміальне інтегрування при $m=0,1,2$

m	$\varphi(x)$	$\int_0^1 \varphi(x)dx$	$\sum_{i=0}^{n} \alpha_i \cdot \varphi(x_i)$
0	$x^0=1$	$\int_0^1 dx = x\Big\|_0^1 = 1$	$\alpha_0 \cdot \left(\frac{1}{4}\right)^0 + \alpha_1 \cdot \left(\frac{1}{2}\right)^0 + \alpha_2 \cdot \left(\frac{3}{4}\right)^0$
1	$x^1=x$	$\int_0^1 xdx = x^2/2\Big\|_0^1 = 1/2$	$\alpha_0 \cdot \left(\frac{1}{4}\right)^1 + \alpha_1 \cdot \left(\frac{1}{2}\right)^1 + \alpha_2 \cdot \left(\frac{3}{4}\right)^1$
2	x^2	$\int_0^1 x^2dx = x^3/3\Big\|_0^1 = 1/3$	$\alpha_0 \cdot \left(\frac{1}{4}\right)^2 + \alpha_1 \cdot \left(\frac{1}{2}\right)^2 + \alpha_2 \cdot \left(\frac{3}{4}\right)^2$

Отримана система (див. останній стовпчик таблиці 10.1) має розв'язок: $\alpha_0=\alpha_2=2/3$, $\alpha_1=-1/3$. Отже, формула поліноміального інтегрування з трьома симетричними квадратурними вузлами має такий вигляд:

$$\int_0^1 \varphi(x)dx \approx \frac{2}{3} f(1/4) - \frac{1}{3} f(1/2) + \frac{2}{3} f(3/4). \qquad (10.6)$$

Можна переконатися, що ця формула дає точні значення навіть для полінома третьої степені.

Аналогічно можна побудувати квадратичні суми для будь-якого значення n та набору квадратурних вузлів.

10.3. Інтерполяційні квадратурні формули

Квадратурними формулами інтерполяційного типу називаються такі, що для визначення квадратурних сум застосовують інтерполяційний поліном Лагранжа. Вони можуть будуватися на нерівномірних або рівномірних сітках. Попередньо визначеними є квадратурні вузли x_i, знаходяться квадратурні коєфіцієнти α_i.

10.3.1. Загальна інтерполяційна квадратурна формула

В інтервалі $[a,b]$ призначається $n+1$ квадратурний вузол x_i (включно з $x_0=a$ та $x_{n+1}=b$, формула закритого типу) з *довільною* відстанню між ними. В якості функції $\varphi(x)$, яка має відому первісну, обирається поліном Лагранжа n-ої степені:

$$\varphi(x) = L_n(x) = \Pi_{n+1}(x) \cdot \sum_{i=0}^{n} \left(\varphi(x_i) \frac{1}{\Pi'_{n+1}(x_i) \cdot (x - x_i)} \right), \qquad (10.7)$$

де поліном

$$\Pi_{n+1}(x) = (x - x_0) \cdot (x - x_1) \cdot \ldots \cdot (x - x_n), \tag{10.8}$$

причому виконана умова $f(x_i) = p(x_i) \cdot \varphi(x_i)$. З урахуванням формул (10.4), (10.7) і (10.8):

$$\int_a^b p(x) \cdot \varphi(x) dx = \int_a^b p(x) \cdot L_n(x) dx + R_n(x) \approx \sum_{i=0}^{n} \alpha_i \varphi(x_i). \tag{10.9}$$

Знак суми та фіксовані значення $\varphi(x_i)$ виносяться за межі інтеграла:

$$\int_a^b p(x) \cdot L_n(x) dx + R_n(x) \approx \sum_{i=0}^{n} \left(\varphi(x_i) \int_a^b p(x) \cdot \frac{\Pi_{n+1}(x)}{\Pi'_{n+1}(x_i) \cdot (x - x_i)} dx \right) = \sum_{i=0}^{n} \alpha_i \varphi(x_i). \tag{10.10}$$

Відповідно до (10.9) з (10.10) вагові коефіцієнти визначаються формулою

$$\alpha_i = \int_a^b p(x) \cdot \frac{\Pi_{n+1}(x)}{\Pi'_{n+1}(x_i) \cdot (x - x_i)} dx. \tag{10.11}$$

Якщо вагові коефіцієнти будуть визначеними з (10.11) точно, то для *многочленів* до степенів n включно похибка $R_n(x)$ буде дорівнювати нулю.

У інших випадках похибка $R_n(x)$ визначається за допомогою формули:

$$R_n(x) = \frac{1}{n!} \int_a^b p(x) \cdot \Pi_{n+1}(x) \cdot \varphi^{(n)}(\xi) dx, \tag{10.12}$$

де $\varphi^{(n)}(\xi)$ – похідна n-ої степені в деякій точці $\xi \in [a, b]$ (якщо така похідна існує).

10.3.2. Квадратурні формули Ньютона-Котеса

В інтервалі $[a,b]$ призначається $n+1$ точка інтегрування (включно з $x_0 = a$ та $x_n = b$, формула закритого типу) з *однаковою* відстанню між ними $h = (b-a)/n$, тобто $x_i = a + hi$. Проводиться заміна: $q = (x - a)/h$, при цьому $x = a + hq$, а $0 \le q \le n$. Вираз (10.8) з урахуванням того, що $b = a + hn$, змінює вигляд на

$$\Pi_{n+1}(x) = \Pi_{n+1}(a + qh) = h^{n+1} \cdot (q - 0) \cdot (q - 1) \cdot \ldots \cdot (q - n). \tag{10.13}$$

Тоді можна одержати, що

$$\Pi'_{n+1}(x_i) = \Pi'_{n+1}(a + ih) = (-1)^{n-i} \cdot h^n \cdot i! \cdot (n - i)! \ . \tag{10.14}$$

Відповідно до виразу (10.11), враховуючи, що $x = a$ при $i = 0$, а $x = b$ при $i = n$ (це – нові межі інтегрування):

$$\alpha_i = \int_0^n p(a + hq) \cdot \frac{(-1)^{n-i}}{i! \cdot (n - i)!} \cdot \frac{q \cdot (q - 1) \cdot \ldots \cdot (q - n)}{(q - i)} h dq. \tag{10.15}$$

Якщо врахувати, що $h = (b - a)/n$, то з (10.15)

$$\alpha_i = (b - a) \cdot \frac{1}{n} \cdot \frac{(-1)^{n-i}}{i! \cdot (n - i)!} \cdot \int_0^n p(a + hq) \cdot \frac{q \cdot (q - 1) \cdot \ldots \cdot (q - n)}{(q - i)} dq. \tag{10.16}$$

При $p(x) \equiv 1$

$$\alpha_i = (b - a) \cdot (H_n)_i, \tag{10.17}$$

де позначені *коефіцієнти Котеса* (Ньютона-Котеса):

$$(H_n)_i = \frac{1}{n} \cdot \frac{(-1)^{n-i}}{i! \cdot (n - i)!} \cdot \int_0^n \frac{q \cdot (q - 1) \cdot \ldots \cdot (q - n)}{(q - i)} dq; \quad i = 0,1 \ldots, n. \tag{10.18}$$

Отже, квадратурна формула Ньютона-Котеса:

$$\int_a^b \varphi(x)dx \approx (b-a)\cdot\sum_{i=0}^{n}(H_n)_i\,\varphi(x_i)\,. \qquad (10.19)$$

Коефіцієнти Ньютона-Котеса мають такі властивості:

$$\sum_{i=0}^{n}(H_n)_i = 1;\quad (H_n)_i = (H_n)_{n-i}\,. \qquad (10.20)$$

Їхні значення поміщено в таблицю 10.2 (прийнято, що $(H_n)_i = (\widetilde{H}_n)_i / w_n$, де w_n – загальний для даного n знаменник).

Таблиця 10.2. Коефіцієнти Ньютона-Котеса

n	$(\widetilde{H}_n)_0$	$(\widetilde{H}_n)_1$	$(\widetilde{H}_n)_2$	$(\widetilde{H}_n)_3$	$(\widetilde{H}_n)_4$	$(\widetilde{H}_n)_5$	$(\widetilde{H}_n)_6$	$(\widetilde{H}_n)_7$	$(\widetilde{H}_n)_8$	w_n
1	1	1								2
2	1	4	1							6
3	1	3	3	1						8
4	7	32	12	32	7					90
5	19	75	50	50	75	19				288
6	41	216	27	272	27	216	41			840
7	751	3577	1323	2989	2989	1323	3577	751		17280
8	989	5888	-928	10496	-4540	10496	-928	5888	989	28350

Квадратурні формули Ньютона-Котеса з непарною кількістю точок інтегрування (n – парні) є точнішими. При $n > 7$ застосовувати формули Ньютона-Котеса не рекомендується, оскільки при $n \geq 8$ з'являються від'ємні коефіцієнти, що може призвести до підвищених похибок.

10.3.3. Окремі випадки квадратурних формул Ньютона-Котеса

Поширеними є три варіанта квадратурних формул Ньютона-Котеса: при $n = 1$ – формула трапецій, при $n = 2$ – формула Сімпсона, при $n = 3$ – формула Ньютона.

10.3.3.1. Формула трапецій

При $p(x) \equiv 1$ та $n = 1$: $x_0 = a$, $x_1 = b$, із таблиці 10.2 $(H_1)_0 = (H_1)_1 = 1/2$. Оскільки $h = b - a$, то з (10.19)

$$\int_a^b \varphi(x)dx \approx (b-a)\cdot[\varphi(a)+\varphi(b)]/2\,. \qquad (10.21)$$

Похибка апроксимації для формули трапеції обчислюється з (10.12) з урахуванням того, що $p(x) \equiv 1$ та $\Pi_{n+1}(x) = (x-a)(x-b)$. Після інтегрування:

$$R(x) = -\frac{h^3}{12}\varphi''(\xi);\ \xi \in [a,b],\ \text{тобто}\ R(x) = O(h^3)\,. \qquad (10.22)$$

При $\varphi''(x) > 0$ формула трапецій дає значення з перебільшенням, при $\varphi''(x) < 0$ – з нестачею.

10.3.3.2. Формула Сімпсона (формула парабол)

При $p(x)\equiv 1$ та $n=2$: $x_0=a$, $x_1=c=(a+b)/2$, $x_2=b$; із таблиці 10.2 $(H_2)_0=(H_2)_2=1/6$, а $(H_2)_1=4/6$. Оскільки $h=(b-a)/2$, то з (10.19)

$$\int_a^b \varphi(x)dx \approx (b-a)\cdot[\varphi(a)+4\varphi(c)+\varphi(b)]/6. \qquad (10.23)$$

Похибка апроксимації для неї обчислюється з формули (10.12) з урахуванням того, що $p(x)\equiv 1$ та $\Pi_{n+1}(x)=(x-a)(x-c)(x-b)$. Після інтегрування:

$$R(x)=-\frac{h^5}{90}\varphi^{(4)}(\xi);\ \xi\in[a,b], \text{ тобто } R(x)=O(h^5). \qquad (10.24)$$

Отже, формула Сімпсона має дуже високий 5-й порядок наближення.

10.3.3.3. Квадратурна формула Ньютона

При $p(x)\equiv 1$ та $n=3$: $x_0=a$, $x_1=c=a+(b-a)/3$, $x_2=e=a+2(b-a)/3$, $x_3=b$, із таблиці 10.2 $(H_3)_0=(H_3)_3=1/8$, а $(H_3)_1=(H_3)_2=3/8$. Оскільки $h=(b-a)/3$, то з (10.19)

$$\int_a^b \varphi(x)dx \approx 3(b-a)\cdot[\varphi(a)+3\varphi(c)+3\varphi(e)+\varphi(b)]/8. \qquad (10.25)$$

Похибка апроксимації для формули Ньютона обчислюється з (10.12) з урахуванням того, що $p(x)\equiv 1$ та $\Pi_{n+1}(x)=(x-a)(x-c)(x-e)(x-b)$. Після інтегрування остаточний вираз:

$$R(x)=-\frac{3h^5}{80}\varphi^{(4)}(\xi);\ \xi\in[a,b], \text{ тобто } R(x)=O(h^5). \qquad (10.26)$$

тобто при однаковому кроці формула Ньютона формально має таку ж точність, як й формула Сімпсона, фактично – десь у два рази меншу ($3/80>1/90$), до того ж потребує більшого обсягу обчислень.

10.3.3.4. Загальні форми квадратурних формул трапецій, Сімпсона та Ньютона

Якщо на інтервалі $[a,b]$ функція $y(x)$ змінюється швидко, або інтервал – значний, то для високої точності інтегрування можна або застосовувати формули Ньютона-Котеса з великим n, або розбити інтервал на декілька інтервалів меншого розміру і на кожному із отриманих інтервалів застосувати формулу Ньютона-Котеса з малим n. У другому випадку, після групування однакових значень функцій, можна отримати *загальні форми* квадратурних формул. Якщо кількість таких інтервалів позначимо як k, то (при $h=(b-a)/k$):

- загальна формула трапецій

$$\int_a^b \phi(x)dx \approx h\cdot[\phi(x_0)/2+\phi(x_1)+\phi(x_2)+\ldots+\phi(x_{k-1})+\phi(x_k)/2], \qquad (10.27)$$

має похибку

$$R(x)=-\frac{h^3}{12\cdot k^2}\varphi''(\xi);\ \xi\in[a,b], \text{ тобто } R(x)=O(h^3). \qquad (10.28)$$

- загальна формула Сімпсона при парній кількості k

$$\int_a^b \varphi(x)dx \approx h\cdot[\varphi(x_0)+4\Sigma_1+2\Sigma_2+\varphi(x_k)]/3, \qquad (10.29)$$

де $\Sigma_1=\varphi(x_1)+\varphi(x_3)+...+\varphi(x_{2m-1})$ та $\Sigma_2=\varphi(x_2)+\varphi(x_4)+...+\varphi(x_{2m})$; має похибку

$$R(x)=-\frac{h^5}{90\cdot k^4}\varphi^{(4)}(\xi);\quad \xi\in[a,b],\ \text{тобто } R(x)=O(h^5). \qquad (10.30)$$

- загальна формула Ньютона при k, кратному трьом

$$\int_a^b \varphi(x)dx \approx 3h\cdot[\varphi(x_0)+2\Sigma_1+3\Sigma_2+\varphi(x_k)]/8, \qquad (10.31)$$

де $\Sigma_1=\varphi(x_3)+\varphi(x_6)+...+\varphi(x_{k-3})$ та $\Sigma_2=\varphi(x_1)+\varphi(x_2)+\varphi(x_4)+\varphi(x_5)+...+\varphi(x_{k-2})+$ $+\varphi(x_{k-1})$; має похибку

$$R(x)=-\frac{3h^5}{80\cdot k^4}\varphi^{(4)}(\xi);\quad \xi\in[a,b],\ \text{тобто } R(x)=O(h^5). \qquad (10.32)$$

Ці формули формально мають такий же порядок точності, як й прості, але фактично – кращий, оскільки у знаменнику з'являється k^2 або k^4 відповідно. При застосуванні формул (10.27) та (10.29) рекомендують провести два розрахунки: з кроком h та $2h$. Співпадаючі десяткові знаки у результатах вважають точними. **Увага**: якщо загальна кількість квадратурних вузлів буде *непарною* та одночасно *кратною трьом*, то загальну формулу Сімпсона застосувати не можливо. Тоді рекомендують використовувати загальну формулу Ньютона.

10.4. Квадратурні формули Чебишева

10.4.1. Алгоритм створення квадратурних формул Чебишева

Квадратурні формули Чебишева отримані при фіксованому (єдиному) ваговому коефіцієнті. Значення координат квадратурних вузлів обрані з умови, що формула є точною для усіх $\varphi(x)$ у вигляді полінома до степені n включно.

Спочатку за допомогою формули

$$x=[a+b+(b-a)q]/2 \qquad (10.33)$$

проводиться заміна аргументу та границь інтегрування, потім – наближення:

$$\int_{-1}^{1} f(q)dq \approx \int_{-1}^{1} p(q)\cdot\varphi(q)dq \approx \alpha\cdot\sum_{i=1}^{n}\varphi(q_i). \qquad (10.34)$$

Оскільки $\alpha=const$ для будь-якого виразу $\varphi(q)$, то й для випадку $\varphi(q)=1$. Тоді з (10.34):

$$\int_{-1}^{1} p(q)\cdot 1\cdot dq=\alpha\cdot\sum_{i=1}^{n}1=\alpha\cdot n,\quad \text{тобто}\quad \alpha=\frac{1}{n}\int_{-1}^{1}p(q)dq. \qquad (10.35)$$

Для отримання значень координат квадратурних вузлів q_i збирається система з n рівнянь за умови, що при $\varphi(q)=q^1, q^2, q^3, ..., q^n$ формула є точною. У загальному вигляді при невідомій ваговій функції $p(q)$ отримати розв'язок системи, тобто знайти q_i, неможливо. Але доведено, що така система матиме єдиний

розв'язок, якщо функція $p(q)$ однозначна в діапазоні $q \in [-1,1]$. **Увага**: значення q_i можуть бути комплексними або не лежати у діапазоні $[-1,1]$.

10.4.2. Квадратурні формули Чебишева при одиничній ваговій функції

Якщо $p(q) \equiv 1$, то з формули (10.35)

$$\int_{-1}^{1} 1 \cdot dq = 2, \quad \text{тобто} \quad \alpha = \frac{2}{n}. \tag{10.36}$$

Отже, формули Чебишева при $p(q) \equiv 1$ мають вигляд

$$\int_{-1}^{1} f(q)dq \approx \int_{-1}^{1} \varphi(q)dq \approx \frac{2}{n} \cdot \sum_{i=1}^{n} \varphi(q_i). \tag{10.37}$$

Для отримання значень q_i збирається система з n рівнянь за умови, що при $\varphi(q) = q^1, q^2, q^3, ..., q^n$ формула є точною. Попередньо можна отримати, що

$$w_1 = \frac{n}{2} \cdot \int_{-1}^{1} q^1 dq = 0; \quad w_2 = \frac{n}{2} \cdot \int_{-1}^{1} q^2 dq = \frac{n}{3}; \quad ...; \quad w_n = \frac{n}{2} \cdot \int_{-1}^{1} q^i dq = \frac{n[1-(-1)^{i+1}]}{2(i+1)}. \tag{10.38}$$

Оскільки з (10.37) $\sum_{i=1}^{n} \varphi(q_i) = \sum_{i=1}^{n} (q_i)^i = w_i$, то система рівнянь буде така:

$$\begin{cases} (q_1)^1 + (q_2)^1 + ... + (q_n)^1 = 0; \\ (q_1)^2 + (q_2)^2 + ... + (q_n)^2 = n/3; \\ \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots; \\ (q_1)^n + (q_2)^n + ... + (q_n)^n = \dfrac{n[1-(-1)^{n+1}]}{2(n+1)}. \end{cases} \tag{10.39}$$

В алгебрі многочленів розроблено алгоритм розв'язування таких нелінійних систем. Спочатку розв'язується лінійна система

$$\begin{cases} w_1 + a_1 = 0; \\ w_2 + a_1 w_1 + 2a_2 = 0; \\ \ldots\ldots\ldots\ldots\ldots\ldots; \\ w_n + a_1 w_{n-1} + a_2 w_{n-2} + ... + n a_n = 0, \end{cases} \tag{10.40}$$

а потім за допомогою отриманих коефіцієнтів a_i будується поліном

$$\Pi_n(q) = q^n + a_1 \cdot q^{n-1} + a_2 \cdot q^{n-2} + ... + a_n, \tag{10.41}$$

корені якого й є координатами квадратурних вузлів q_i.

У нашому випадку, оскільки все непарні $w_1 = w_3 = ... = 0$, то й всі непарні $a_1 = a_3 = ... = 0$. Парні коефіцієнти знаходяться із залишків системи (10.40):

$$\begin{cases} n/3 + 2a_2 = 0; \\ n/5 + n a_2/3 + 4a_4 = 0; \\ n/7 + n a_2/5 + n a_4/3 + 6a_6 = 0; \\ \ldots\ldots\ldots\ldots\ldots\ldots\ldots \end{cases} \tag{10.42}$$

Розглянемо деякі окремі випадки.

При $n=1$, ваговий коефіцієнт $\alpha=2$ та $a_1=0$. Тому з (10.41) отримаємо, що $\Pi_1(q)=q+a_1=q$. Величина q_1 знаходиться з рівняння $\Pi_1(q_1)=0$, тому й $q_1=0$. Отже, при $n=1$ формула Чебишева:

$$\int_{-1}^{1} f(q)dq \approx \int_{-1}^{1} \varphi(q)dq \approx 2\cdot\varphi(0). \qquad (10.43)$$

При $n=2$, ваговий коефіцієнт $\alpha=1$ та, оскільки $w_1=0$ і $w_2=2/3$, з (10.40) маємо $a_1=0$, $a_2=-1/3$. Тому з (10.41) отримаємо, що $\Pi_1(q)=q^2+a_2$. Величини q_1 та q_2 знаходяться з рівняння $\Pi_1(q)=q^2+a_2=0$, тому $q_1=-1/\sqrt{3}$ та $q_2=1/\sqrt{3}$. Отже, при $n=2$ формула Чебишева:

$$\int_{-1}^{1} f(q)dq \approx \int_{-1}^{1} \varphi(q)dq \approx \varphi(-1/\sqrt{3})+\varphi(1/\sqrt{3}). \qquad (10.44)$$

Отримані подібнім чином значення q_i поміщені в таблицю 10.3. Враховано, що при $n=8$ та $n\geq 10$ система (10.42) не має дійсних розв'язків.

Таблиця 10.3. Значення вагових коефіцієнтів та координат квадратурних вузлів квадратурних формул Чебишева

n	α	i	q_i
1	2	1	0
2	1	1, 2	∓0.577356
3	2/3	1, 3	∓0.707107
		2	0
4	1/2	1, 4	∓0.794654
		2, 5	∓0.187592
5	2/5	1, 5	∓0.832498
		2, 4	∓0.374541
		3	0
6	1/3	1, 6	∓0.86625
		2, 5	∓0.42252
		3, 4	∓0.26664
7	2/7	1, 7	∓0.88386
		2, 6	∓0.52966
		3, 5	∓0.32391
		4	0
9	2/9	1, 9	∓0.91159
		2, 8	∓0.60102
		3, 7	∓0.52876
		4, 6	∓0.16791
		5	0

Примітка 10.1. З теорії імовірності випливає, що застосування рівних вагових коефіцієнтів мінімізує імовірнісну похибку, якщо значення $\varphi(q)$ відповідають нормальному закону розподілу випадкових похибок.

10.5. Квадратурні формули найвищої алгебраїчної степені точності

10.5.1. Загальні положення

Застосовується загальний підхід: значення координат квадратурних вузлів та вагових коефіцієнтів заздалегідь невідомі; формула повинна бути точною для всіх $\varphi(x)$ у вигляді полінома до степені n включно.

Із застосуванням формули (10.33) проводиться заміна аргументу та границь інтегрування, тому

$$\int_{-1}^{1} f(q)dq \approx \int_{-1}^{1} p(q)\cdot\varphi(q)dq \approx \sum_{i=1}^{n}\alpha_i\cdot\varphi(q_i)\,. \qquad (10.45)$$

Доведено теорему, що для того, щоб формула (10.45) була точною для всіх многочленів степені $2n-1$ включно, необхідно та достатньо, щоб ця формула була інтерполяційною, функція $p(q)$ не змінювала знак, а також щоб квадратурні вузли були обрані так, щоб многочлен $\Pi_n(q)=(q-q_1)\cdot(q-q_2)\cdot\ldots\cdot(q-q_n)$ був ортогональним до кожного многочлена $Q_k(q)$, степінь якого менша, ніж n:

$$\int_{-1}^{1}\Pi_n(q)\cdot Q_k(q)dq=0\,;\quad k=0,1,\ldots<n=1,2,\ldots\,. \qquad (10.46)$$

Тоді всі корені цього многочлена – різні та належать діапазону $[-1,1]$. Якщо функція $p(q)\ge 0$, то всі квадратурні коефіцієнти є додатними. Також показано, що похибка $R_n(q)$ наближеного інтегрування при наявності безперервних похідних дається формулою

$$R_n(x)=\frac{1}{(2n)!}\cdot\varphi^{(2n)}(\xi)\cdot\int_{-1}^{1}p(q)\cdot(\Pi_n(q))^2\cdot dq\,, \qquad (10.47)$$

де $\varphi^{(2n)}(\xi)$ – похідна вказаної степені в деякій точці $\xi\in[a,b]$ (якщо вона існує).

10.5.2. Квадратурні формули Гаусса (Гаусса-Лежандра)

Вважається, що вагова функція $p(q)\equiv 1$. У квадратурних формулах Гаусса в якості полінома $\Pi_n(q)$ використовується поліном Лежандра

$$L_n(q)=\frac{1}{2^n n!}\frac{d^n}{dq^n}(q^2-1)^n\,;\quad n=0,1,\ldots, \qquad (10.48)$$

який в інтервалі $[-1,1]$ має такі властивості:

- значення на краях інтервалу $L_n(1)=1$; $L_n(-1)=(-1)^n$; $n=0,1,\ldots$;
- має n різних дійсних коренів;
- ортогональний до будь-якого полінома $Q_k(q)$ при $k<n$, тобто виконується умова (10.46).

Для визначення величин α_i та q_i потрібно зібрати систему з $2N$ рівнянь, де N – найбільше значення n. Перші N рівнянь для визначення квадратурних

вузлів q_i утворюються як вирази коренів многочлена (10.48). Ще N рівнянь для визначення квадратурних коефіцієнтів утворюються як результати аналітичного інтегрування (10.45) при $\varphi(q) = L_n(q)$, $n = 1,2,...,N$. Розв'язок отриманих систем рівнянь – значення величин α_i та q_i, поміщено в таблицю 10.4. Крім того, є явна формула для визначення α_i:

$$a_i = \frac{2 \cdot [1 - (q_i)^2]}{n^2 \cdot [L_{n-1}(q_i)]^2}. \tag{10.49}$$

Таблиця 10.4. Значення вагових коефіцієнтів та координат квадратурних вузлів квадратурної формули Гаусса (Гаусса-Лежандра)

N	i	α_i	q_i
1	1	2	0.0
2	1, 2	1	$\mp\sqrt{1/3}$
3	1, 3	5/9	$\mp\sqrt{3/5}$
	2	8/9	0.0
4	1, 4	0.347854845137454	∓ 0.861136311594053
	2, 3	0.652145154862546	∓ 0.339981043584856
5	1, 5	0.236926885056189	∓ 0.906179845938664
	2,4	0.478628670499366	∓ 0.538469310105683
	3	0.568888888888889	0.0
6	1, 6	0.171324492379170	∓ 0.932469514203152
	2, 5	0.360761573048139	∓ 0.661209386466265
	3, 4	0.467913934572691	∓ 0.238619186083197
7	1, 7	0.129484966168870	∓ 0.949107912342759
	2, 6	0.279705391489277	∓ 0.741531185599394
	3, 5	0.381830050505119	∓ 0.405845151377397
	4	0.417959183673469	0.0
8	1, 8	0.101228536290376	∓ 0.960289856497536
	2, 7	0.222381034453374	∓ 0.796666477413627
	3, 6	0.313706645877887	∓ 0.525532409916329
	4, 5	0.362683783378362	∓ 0.183434642495650
9	1, 9	0.081274388361574	∓ 0.968160239507626
	2, 8	0.180648160694857	∓ 0.0836031107326636
	3, 7	0.260610696402935	∓ 0.613371432700590
	4, 6	0.312347077040003	∓ 0.324253423403809
	5	0.330239355001260	0.0
10	1, 10	0.066671344308688	∓ 0.973906528517172
	2, 9	0.149451349150581	∓ 0.865063366688985
	3, 8	0.219086362515982	∓ 0.679409568299024
	4, 7	0.269266719309996	∓ 0.433395394129247
	5, 6	0.295524224714753	∓ 0.148874338981631

Формула є точною для многочленів степені 1 при $N = 1$; 3 при $N = 2$; 5 при $N = 3$, тощо.

10.5.3. Про квадратурні формули Гаусса-Лобатто (Маркова)

Фіксуються координати квадратурних вузлів на границі діапазону [-1,1], а в якості полінома $\Pi_n(q)$ використовується поліном Лежандра (10.48). Якщо застосувати 3 вузла, то отримаємо формулу Ньютона-Котеса (див. табл.10.2 при $n=2$). Якщо застосувати 4 вузла, то для формули (10.45) отримаємо, що $q_4=-q_1=1;\ q_3=-q_2=1/\sqrt{5};\ \alpha_4=\alpha_1=1/6;\ \alpha_3=\alpha_2=5/6$, причому формула є точною для многочленів степені п'ять.

10.5.4. Про квадратурні формули найвищої алгебраїчної степені точності для деяких вагових функцій

Відомі декілька квадратурних формул найвищої алгебраїчної степені точності для вагових функцій, що не дорівнюють одиниці. Наприклад:

- $p(x)=(b-x)^{\alpha}(x-a)^{\beta}$ при $\alpha,\beta>-1$ та $a<x<b$ (вагова функція Якобі). Описує особливості функції $f(x)$ на кінцях відрізку $[a,b]$;
- $p(x)=x^{\alpha}e^{-x}$ при $\alpha>-1$ та $0<x<\infty$ (вагова функція Лагерра). Описує особливості функції $f(x)$ в точці $x=0$, а також швидкість убування при $x\to\infty$;
- $p(x)=e^{-x^2}$ при $-\infty<x<\infty$ (вагова функція Ерміта). Описує швидкість убування функції $f(x)$ при $x\to-\infty$ та $x\to\infty$.

Докладно про ці випадки викладено, наприклад, у книзі [34] зі стор. 251.

10.6. Завершення

Окрім описаних вище варіантів наближеного обчислення інтегралів є ще й інші, зокрема:

- наближене інтегрування інтегралів з урахуванням заздалегідь призначених вузлів інтегрування;
- наближене інтегрування невласних інтегралів;
- наближене інтегрування невизначених інтегралів;
- наближене інтегрування інтегралів із застосуванням сплайнів.

Крім того існує проблема підвищення точності наближеного обчислення інтегралів, яка вирішується не тільки збільшенням значення степені многочлена n або кількості інтервалів, але й шляхом додавання до квадратурної суми нових членів, які повністю або в основному наближують остаточний член квадратурної формули $R_n(x)$.

Контрольні питання до підрозділу 10.1

1. Які є три варіанти попередньо визначення квадратурних коефіцієнтів та вузлів при наближеному інтегруванні функцій?
2. Які формули чисельного інтегрування називають квадратурними?

Контрольні питання до підрозділу 10.2

1. Чому поліноміальне інтегрування не може мати значну точність?

Контрольні питання до підрозділу 10.3

1. Які принципові особливості мають квадратурні формули інтерполяційного типу?
2. Яку кількість квадратурних вузлів (парну або непарну) має загальна інтерполяційна квадратурна формула?
3. Чи може загальна інтерполяційна квадратурна формула мати нульову похибку інтегрування?
4. Які властивості мають коефіцієнти Ньютона-Котеса?
5. Які відомі формули наближеного інтегрування є окремими випадками формул Ньютона-Котеса?

Контрольні питання до підрозділу 10.4

1. Чи є обмеження в підвищенні точності інтегрування за квадратурними формулами Чебишева за рахунок підвищення степені n полінома, що наближує?

Контрольні питання до підрозділу 10.5

1. Який загальний підхід застосовується для створення квадратурних формул найвищої алгебраїчної степені точності?
2. Який поліном застосовується для отримання квадратурних формул Гаусса-Лежандра? Яка важлива властивість цього полінома використовується при цьому?