Б. И. Квасов

ЧИСЛЕННЫЕ МЕТОДЫ АНАЛИЗА И ЛИНЕЙНОЙ АЛГЕБРЫ

ФЕДЕРАЛЬНОЕ АГЕНТСТВО ПО ОБРАЗОВАНИЮ
НОВОСИБИРСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ

Механико-математический факультет

Б. И. Квасов

ЧИСЛЕННЫЕ МЕТОДЫ АНАЛИЗА И ЛИНЕЙНОЙ АЛГЕБРЫ

Учебное пособие

Новосибирск
2012

ББК В192.14я73-1
УДК 519.6 (075)
К 324

Квасов Б.И. Численные методы анализа и линейной алгебры. Учеб. пособие / Новосиб. гос. ун-т. Новосибирск, 2012. 262 с.

ISBN 978-5-94356-709-4

В учебном пособии излагается ряд тем классического курса численного анализа: решение нелинейных уравнений, интерполяция многочленами Лагранжа и сплайнами, метод наименьших квадратов и сплайн-сглаживание, численное дифференцирование и интегрирование. Рассмотрены две основные задачи численных методов линейной алгебры: решение систем линейных уравнений и отыскание собственных значений и собственных векторов матриц. Основная цель пособия – помочь студентам в освоении современных численных методов, изложив их в наиболее простой и доступной форме. С этой целью изложение иллюстрируется примерами и сопровождается задачами для самостоятельной работы студентов. Дается краткое введение в интерактивную систему Матлаб, позволяющую организовать эффективный компьютерный практикум по численным методам, и описание семи лабораторных работ. Приведены тесты для письменного экзамена по основам численных методов.

Предназначено для студентов математических и физических специальностей университетов, технических вузов и колледжей.

Пособие подготовлено в рамках реализации Программы развития НИУ-НГУ.

Рецензенты:
канд. физ.-мат. наук, доц. А. П. Михайлов
д-р физ.-мат. наук, проф. Г. С. Хакимзянов
д-р физ.-мат. наук, проф. С. П. Шарый

Рекомендовано редакционно-издательским советом НГУ для специальностей 0647, 2013.

ISBN 978-5-94356-709-4

ОГЛАВЛЕНИЕ

«... эффект, достигаемый за счет совершенствования численных методов, по порядку сравним с эффектом, достигаемым за счет повышения производительности ЭВМ». Н. С. Бахвалов и др. «Численные методы»

Предисловие

В течении многих лет автор читает лекции, проводит семинарские занятия и руководит компьютерным практикумом по численным методам на механико-математическом и физическом факультетах Новосибирского госуниверситета и в Высшем колледже информатики НГУ. Данное пособие отражает накопленный опыт преподавания курса численных методов и соответствует программе курса численных методов в НГУ. Выбор материала сделан с учетом программы курса «Вычислительные методы линейной алгебры», читаемого на механико-математическом факультете НГУ. Большое внимание уделено примерам, иллюстрирующим применение различных вычислительных процедур, с тем, чтобы облегчить студентам понимание теоретического материала и привить им навыки квалифицированного использования численных методов.

Книга состоит из двух частей: численного анализа (гл. 1–4) и собственно вычислительных методов линейной алгебры (гл. 5 и 6), содержит три приложения с кратким введением в интерактивную систему Матлаб, описанием лабораторных работ и тестами для письменного экзамена, библиографический список.

В гл. 1 рассматриваются численные методы решения нелинейных уравнений, которые возникают во многих физических задачах. Излагаются основные методы решения таких уравнений, проводится анализ их сходимости, дается их геометрическая интерпретация. Методы сравниваются по скорости их сходимости. Хотя большинство численных методов решения одного нелинейного уравнения могут быть интерпретированы как частные случаи метода простой итерации, последний излагается в конце главы с тем, чтобы вначале познакомить студентов с методами, наиболее распространенными на практике.

Гл. 2 дает описание интерполяции классическими многочленами Лагранжа и Ньютона. Приводится обобщенная схема Горнера, позволяющая вычислять значения интерполяционного многочлена Ньютона и его производных за минимальное число арифметических операций. Так как между узлами интерполяции многочлены Лагранжа могут очень сильно колебаться, то для предотвращения этого

явления на практике используются их кусочные аналоги, называемые лагранжевыми сплайнами. Подробно изучаются такие употребительные в приложениях лагранжевы сплайны как кусочно-линейная интерполяция и кусочно-кубические многочлены Лагранжа. Показано, как с помощью простых приемов можно получить гладкие аналоги лагранжевых сплайнов, известные как локально-аппроксимационные сплайны. Центральное место в главе занимает описание алгоритмов построения интерполяционных кубических сплайнов, играющих важнейшую роль в приложениях.

В гл. 3 излагается метод наименьших квадратов (МНК) получения аппроксимаций в случае задания неточных данных. Рассмотрены нормальные системы МНК на основе как многочленов, так и базисных сплайнов. Последние применяются при большом числе данных, когда затруднительно использовать для описания данных одну формулу. На практике часто возникает задача проведения кривой в заданном коридоре ошибок при известной допустимой погрешности исходных данных. Для решения этой задачи применяется сглаживающий сплайн, который возникает при регуляризации МНК. Дано описание алгоритмов построения сглаживающих сплайнов, включая выбор весовых множителей и параметра регуляризации (сглаживания).

В гл. 4 изучаются методы численного дифференцирования и интегрирования, которые не только важны сами по себе, но и существенно используются при конструировании методов численного решения дифференциальных и интегральных уравнений. Здесь рассматриваются только наиболее употребительные из этих методов, позволяющие понять основные принципы их построения.

Гл. 5 посвящена численному решению систем линейных алгебраических уравнений. В ней излагается основной аппарат численных методов линейной алгебры; дается понятие плохой обусловленности системы линейных уравнений и выводятся оценки числа обусловленности; подробно изучается гауссово исключение, включая его матричную формулировку и алгоритм с выбором ведущих элементов. Для симметрических положительно определенных матриц изложен метод квадратных корней (Холесского); описаны особенности его работы для разреженных матриц. Исследовано поведение числа обусловленности при матричных преобразованиях и показана несостоятельность гауссова исключения. Подробно излагаются методы, основанные на ортогональных преобразованиях: вращений, ортогонализации Грама-Шмидта, отражений. Рассмотрено использование ортогональных преобразований при решении задач метода наименьших квадратов. Для подготовки линейных систем к итерациям предлагается использовать предобуславливатели; описана специфика таких преобразований. Изучаются итерационные методы Якоби и Зейделя, верхней релаксации, методы простой итерации и Ричардсона, метод наискорейшего градиентного спуска; исследуется сходимость

этих методов. Для решения плохообусловленных систем предлагается использовать метод регуляризации.

В гл. 6 излагаются методы решения задач на собственные значения. Вначале изучается устойчивость задачи на собственные значения; затем рассматриваются классический степенной метод а также обратный степенной метод со сдвигами и метод исчерпывания. Для отыскания всех собственных значений и собственных векторов симметрических матриц средней размерности предлагается метод вращений Якоби; доказывается его сходимость. Для нахождения собственных значений произвольной квадратной матрицы предлагается предварительно привести ее к верхней форме Хессенберга (или к трехдиагональной форме в случае ее симметричности) с помощью вращений Гивенса или используя метод отражений Хаусхолдера. Далее применяется классический QR-алгоритм со сдвигами. Для приведения симметрической матрицы к трехдиагональной форме предлагается также использовать метод Ланцоша, реализуемый по явным формулам. Этот метод эффективнее применения матриц вращения и отражения. В шестой главе дается понятие сингулярного разложения произвольной прямоугольной матрицы.

В конце каждой главы приводятся задачи, которые могут быть использованы как для проведения семинарских занятий, так и для самостоятельной работы студентов. Для большинства задач приведены ответы и комментарии, а для наиболее трудных – подробные решения.

В приложении А дается краткое описание интерактивной системы Матлаб, позволяющей организовать весьма эффективный компьютерный практикум по методам вычислений. Практикум является неотъемлемой частью курса методов вычислений и должен привить студентам навыки их практического использования. В приложении Б дано описание семи лабораторных работ по решению нелинейных уравнений, интерполяции, методу наименьших квадратов и сплайн-сглаживанию, численному интегрированию, решению систем линейных уравнений, нахождению собственных значений и собственных векторов матриц. Тесты для письменного экзамена в приложении В могут быть использованы на экзамене по курсу в качестве дополнительного вопроса к основному билету, содержащему, как правило, вопросы по теории численных методов.

Интерактивные технологии очень важны для эффективного обучения. Однако чудеса техники еще не гарантируют эффективного интерактивного обучения: необходима методическая поддержка и повышение квалификации преподавателей. Данное пособие призвано в какой-то степени восполнить этот пробел.

Автор выражает признательность профессору Г. С. Хакимзянову, доценту А. П. Михайлову и д-ру физ.-мат. наук С. П. Шарому, внимательно прочитавшим рукопись и сделавшим ряд полезных замечаний, позволивших улучшить изложение материала.

Глава 1
Решение нелинейных уравнений

В этой главе мы рассмотрим задачу об отыскании решений нелинейного уравнения

$$f(x) = 0, \quad a \le x \le b, \tag{1.1}$$

где функция f является непрерывной или более гладкой. Такая задача возникает, в частности, при математическом моделировании проблем естествознания. Решения задачи (1.1) называются *корнями* уравнения (1.1) или *нулями* функции f. В общем случае решения задачи (1.1) не могут быть найдены точно. Поэтому рассмотрим численные методы нахождения этих решений. Обсуждаемые в главе методы являются итерационными. Они делятся на два вида: методы, сходимость которых гарантирована, и методы, сходимость которых существенно зависит от правильного выбора начального приближения.

§ 1.1. Задача о погружении шара

Вначале рассмотрим пример физической задачи, приводящей к уравнению вида (1.1). Изучим задачу о погружении сферического шара радиуса r в воду. Предположим, что шар сделан из материала, плотность ρ которого меньше плотности воды. Например, шар может быть сделан из дерева. Следовательно, только часть шара окажется под водой, а сам шар будет плавать. Нас интересует на какую глубину d погрузится шар.

Масса воды M_B, вытесненной шаром при его погружении на глубину d, вычисляется по формуле

$$M_B = \int_0^d \pi\left[r^2 - (x-r)^2\right] dx = \frac{\pi d^2(3r-d)}{3}.$$

Здесь площадь горизонтального сечения шара $S = \pi(r^2 - (x-r)^2)$ интегрируется в пределах от 0 до d (рис. 1.1). Масса самого шара находится как $M_{\text{Ш}} = 4\pi r^3\rho/3$. Согласно закону Архимеда масса вытесненной шаром воды должна быть равна массе шара, т. е. $M_B = M_{\text{Ш}}$, что дает нам уравнение

$$\frac{\pi}{3}(d^3 - 3d^2r + 4r^3\rho) = 0.$$

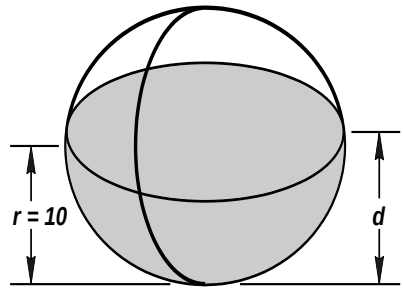


Рис. 1.1. Часть шара радиуса r, погруженная в воду на глубину d

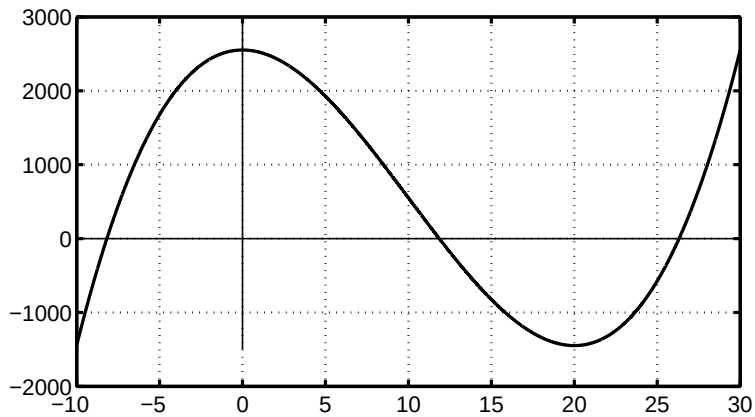


Рис. 1.2. График кубического многочлена $y = 2552 - 30d^2 + d^3$

Предположим, что $r = 10$ и $\rho = 0{,}638$. В этом случае приходим к уравнению

$$f(d) = \frac{\pi}{3}(2552 - 30d^2 + d^3) = 0.$$

График кубического многочлена $y = 2552 - 30d^2 + d^3$ показан на рис. 1.2. Легко видеть, что его корни d_1, d_2 и d_3 лежат соответственно на отрезках $[-10, -5]$, $[10, 15]$ и $[25, 30]$. Первый корень d_1 не представляет для нас интереса, так как является отрицательным. Третий корень d_3 дает значение больше диаметра шара и также должен быть отброшен. Второй корень $d_2 \approx 12$ является искомым решением, когда большая часть шара оказывается погруженной в воду.

Таким образом, рассмотренная задача о погружении шара свелась к поиску корней нелинейного уравнения вида (1.1).

§ 1.2. Отделение корней

Численное решение задачи (1.1) обычно разбивается на два этапа:

1. Отделение корней, т. е. нахождение подотрезков $[\alpha, \beta]$, содержащих в точности один нуль функции f.

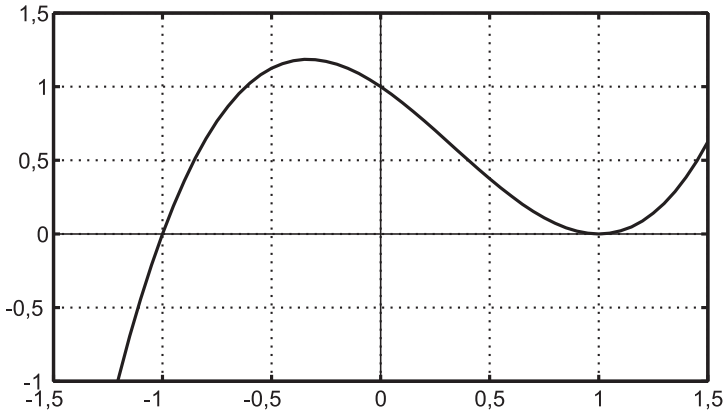


Рис. 1.3. График кубического многочлена $f(x) = (x+1)(x-1)^2$

2. Уточнение корней, когда корни требуется вычислить с заданной точностью, т. е. с заданным числом правильных десятичных знаков.

Для отделения корней обычно находят значения функции $f(x_i)$, где $x_i = a+ih$, $i = 0, 1, \ldots, N$, с некоторым малым шагом $h = (b-a)/N$. Затем выбирают такие подотрезки $[x_i, x_{i+1}]$, что $f(x_i)f(x_{i+1}) < 0$ и, следовательно, на этих подотрезках непрерывная функция f имеет нули. Этот алгоритм не позволяет отделить кратные корни, т. е. точки x^* такие, что $f(x^*) = f'(x^*) = 0$, когда график функции f касается оси x, но не пересекает ее.

Пример 1.1. Рассмотрим функцию $f(x) = (x+1)(x-1)^2$, $|x| \le 1{,}5$, график которой приведен на рис. 1.3. В точке $x^* = 1$ функция f имеет кратный нуль, так как $f(1) = f'(1) = 0$. График функции f касается оси абсцисс, но не пересекает ее. Следовательно, при отделении корней нужно также исследовать точки перемены знака производной f', т. е. точки экстремума.

В дискретной постановке, если разность $\Delta f(x_j) = f(x_j + h) - f(x_j)$ меняет знак в точке x_i и значение функции $f(x_i)$ достаточно мало, то можно считать, что кратный нуль функции f принадлежит подотрезку $[x_{i-1}, x_{i+1}]$.

Таким образом, алгоритм отделения корней уравнения (1.1) может основываться на определении участков монотонности функции f и состоять из двух частей:

а) отделение простых корней, т. е. нахождение отрезков $[x_i, x_i + h]$ таких, что $f(x_i)f(x_i + h) \le 0$.

б) отделение кратных корней, т. е. нахождение отрезков $[x_{i-1}, x_{i+1}]$ таких, что $\Delta f(x_i - h)\Delta f(x_i) \le 0$ и $|f(x_i)| < \varepsilon$ для некоторого малого $\varepsilon > 0$.

Пример 1.2. Отделим нули функции $f(x) = x^4 - x - 1$. Здесь уравнение $f'(x) = 0$ имеет единственный корень $x^* \approx 0{,}36$. Так как $f(-\infty) > 0$, $f(x^*) < 0$, $f(\infty) > 0$ и $(x - x^*)f'(x) > 0$ для $x \ne x^*$, то функция f монотонно убывает при $x < x^*$ и монотонно возрастает при $x > x^*$. Она имеет два нуля, которые

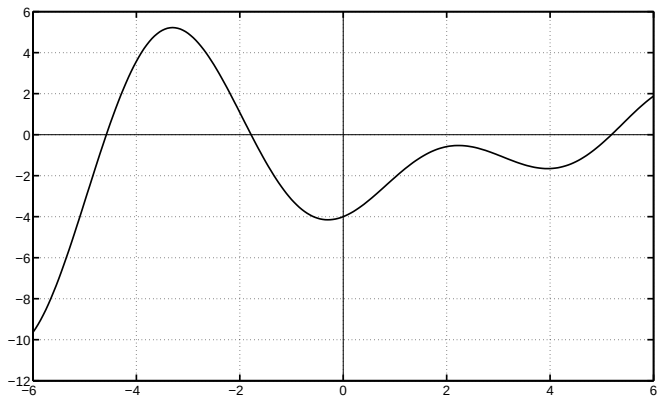


Рис. 1.4. График функции $f(x) = (x-3)cos(x) - 1$ на отрезке $[-\pi, \pi]$

находятся на отрезках $[-1, 0]$ и $[1, 2]$.

§ 1.3. Метод деления отрезка пополам (метод проб)

Будем считать, что уравнение (1.1) имеет на $[a, b]$ единственный корень и $f(a)f(b) < 0$. Для его уточнения рассмотрим итерационный метод деления отрезка пополам, называемый также методом проб.

Разделим отрезок $[a, b]$ пополам и положим $c = (a + b)/2$. Если $f(c) = 0$, то корень найден. При $f(a)f(c) < 0$ в качестве нового отрезка $[a_1, b_1]$ выбираем отрезок $[a, c]$ иначе $[c, b]$. Новый отрезок $[a_1, b_1]$ опять делим пополам и выполняем те же действия. В результате получаем точное значение корня или последовательность отрезков $[a_i, b_i]$, $i = 1, 2, \ldots, n$. Вычисления прекращаем, когда

$$b_n - a_n = \frac{b-a}{2^n} < \varepsilon,$$

где ε – заданная точность нахождения корня.

Пример 1.3. Пусть требуется отделить корни уравнения

$$f(x) = (x-3)\cos(x) - 1 = 0, \quad -2\pi \le x \le 2\pi$$

и уточнить один из них с точностью до 0,01.

Вначале графически локализуем корни. На рис. 1.4 видно, что на отрезке $[-6, 6]$ функция f имеет три нуля. Уточним нуль, находящийся на отрезке $[4, 6]$. Используем метод деления отрезка пополам. На рис. 1.5 «паучок» иллюстрирует работу метода проб, когда в качестве начального приближения берется точка $x_0 = 6$. После 8 итераций $(b-a)/2^8 < 0,01$ и в качестве приближенного значения корня берется величина $x^* \approx 5{,}18$.

Метод деления отрезка пополам всегда сходится, но сходимость его довольно медленная. Этот метод не может быть использован для поиска кратных корней,

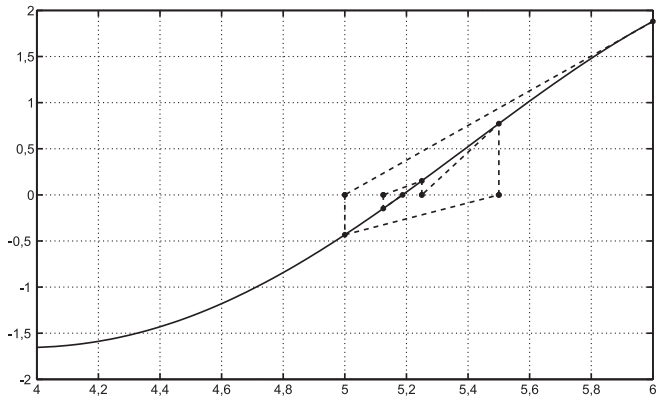


Рис. 1.5. Решение уравнения $f(x) = (x-3)cos(x) - 1 = 0$ по методу проб. Черными точками обозначены получаемые приближения и значения функции f в этих точках

когда $f(a)f(b) > 0$, и не обобщается на случай решения системы нелинейных уравнений.

§ 1.4. Метод хорд (метод ложного положения)

Пусть $f(a)f(b) < 0$. Если $f(a)f''(x) > 0$ для всех $x \in [a,b]$, то на отрезке $[a,b]$ функция f возрастает и выпукла вверх или убывает и выпукла вниз. Запишем уравнение прямой, проходящей через точки $(a, f(a))$ и $(x_k, f(x_k))$, и найдем точку пересечения этой прямой с осью абсцисс. В результате приходим к итерационной формуле

$$x_{k+1} = x_k - \frac{f(x_k)(x_k - a)}{f(x_k) - f(a)}, \quad x_0 = b, \quad k = 0, 1, \ldots . \tag{1.2}$$

Здесь фиксирована точка $(a, f(a))$ (рис. 1.6). Формула (1.2) может быть также переписана в виде

$$x_{k+1} = a - \frac{f(a)(x_k - a)}{f(x_k) - f(a)}, \quad x_0 = b, \quad k = 0, 1, \ldots .$$

Если $f(a)f''(x) < 0$ для всех $x \in [a,b]$, то на отрезке $[a,b]$ функция f возрастает и выпукла вниз или убывает и выпукла вверх. В этом случае фиксированной будет точка $(b, f(b))$. В качестве начального приближения нужно взять $x_0 = a$ и в итерационной формуле (1.2) a заменить на b.

Итерации прекращаем при выполнении условия

$$|x_{k+1} - x_k| < \varepsilon \quad \text{или} \quad \frac{|x_{k+1} - x_k|}{|x_k|} < \varepsilon, \tag{1.3}$$

где ε – требуемая точность вычислений.

Можно также воспользоваться теоремой Лагранжа. Если x^* – корень, то

$$f(x^r) - f(x^*) = f'(\xi)(x^k - x^*), \quad \xi \in (x^k, x^*).$$

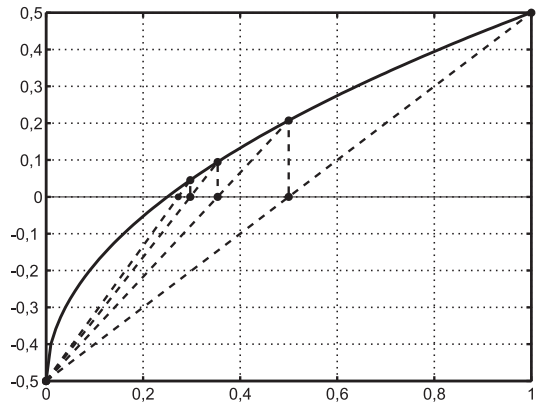


Рис. 1.6. Итерации по методу хорд для уравнения $f(x) = \sqrt{x} - 0{,}5 = 0$

Отсюда

$$|x^k - x^*| \le \frac{|f(x^k)|}{\inf |f'(x)|}. \tag{1.4}$$

Это и есть критерий остановки. Величину $\inf |f'(x)|$ находим аналитически или на сетке значений функции f.

§ 1.5. Метод Ньютона (метод касательных или линеаризации)

Воспользуемся разложением функции f в точке x_k по формуле Тейлора

$$0 = f(x) = f(x_k) + f'(x_k)(x - x_k) + O((x - x_k)^2).$$

Отбрасывая здесь остаточный член, линеаризуем полученное разложение и в предположении $f'(x_k) \neq 0$ приходим к формуле

$$x_{k+1} = x_k - \frac{f(x_k)}{f'(x_k)}, \quad k = 0, 1, \ldots . \tag{1.5}$$

Геометрически данная формула означает, что в точке $(x_k, f(x_k))$ проводится касательная прямая к графику функции f до ее пересечения с осью x в точке x_{k+1}.

Итерации прекращаем при выполнении одного из условий (1.3) или (1.4).

Пример 1.4. Пусть требуется найти $\sqrt[n]{a}$. Положим $f(x) \equiv x^n - a = 0$. Тогда итерации по Ньютону имеют вид:

$$x_{k+1} = x_k - \frac{x_k^n - a}{n x_k^{n-1}} = \frac{1}{n}\left[(n-1)x_k + \frac{a}{x_k^{n-1}}\right], \quad k = 0, 1, \ldots .$$

Сходимость метода Ньютона существенно зависит от правильного выбора начального приближения. При неудачном выборе начального приближения метод

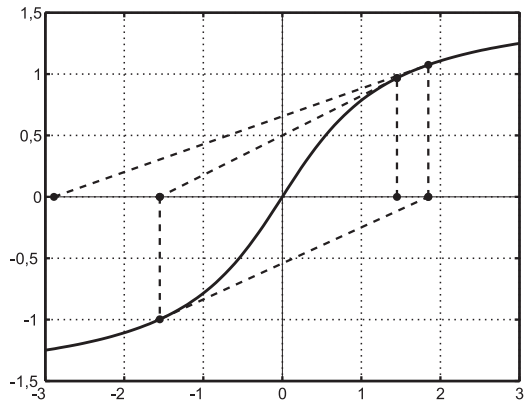


Рис. 1.7. Расходимость метода Ньютона для функции $f(x) = arctan(x)$ при $x_0 = 1{,}45$

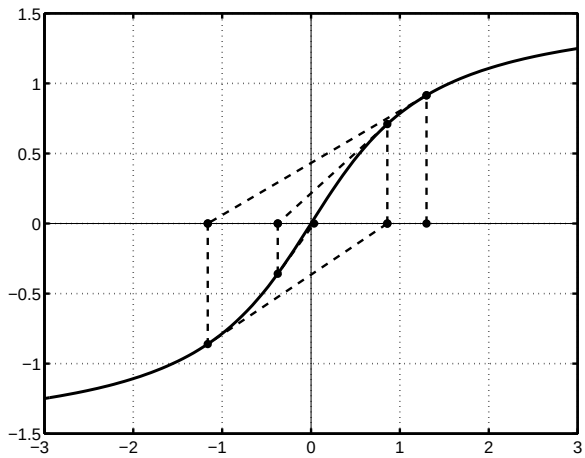


Рис. 1.8. Сходимость метода Ньютона для функции $f(x) = arctan(x)$ при $x_0 = 1{,}3$

может расходиться или даже циклить. В качестве примера рассмотрим решение уравнения $f(x) = arctan(x) = 0$. Здесь корень уравнения $x^* = 0$. Если в качестве начального приближения берется точка $x_0 = 1{,}45$, то метод расходится (рис. 1.7):

$$x_1 = -1{,}55, \quad x_2 = 1{,}85, \quad x_3 = -2{,}89, \quad \ldots$$

Однако если в качестве начального приближения взять более близкую к корню точку $x_0 = 1{,}3$, то метод сходится (рис. 1.8):

$$x_1 = -1{,}16, \quad x_2 = 0{,}85, \quad x_3 = -0{,}37, \quad x_4 = 0{,}03, \quad x_5 = 0{,}00.$$

Ситуация с зацикливанием метода Ньютона рассмотрена на рис. 1.9. Здесь ищется корень уравнения $f(x) = x^3 - x - 3 = 0$. В качестве начального приближения берется точка $x_0 = 0$. Итерации Ньютона дают:

$$x_1 = -3{,}00, \quad x_2 = -1{,}96, \quad x_3 = -1{,}14, \quad x_4 = -0{,}00,$$

после чего происходит зацикливание: $x_{k+4} \approx x_k$, $k = 0, 1, \ldots,$. При $x_0 = 2$ метод сходится к корню $x^* \approx 1{,}67$.

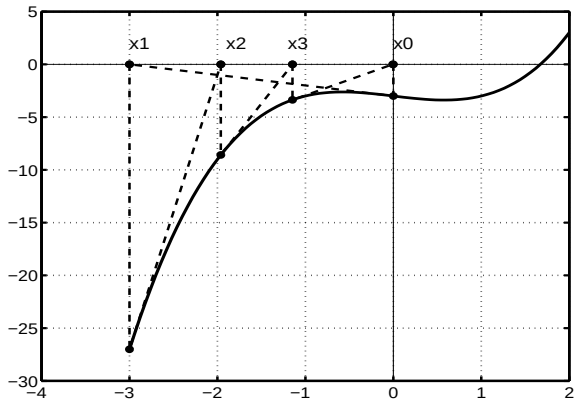


Рис. 1.9. Зацикливание метода Ньютона для уравнения $f(x) = x^3 - x - 3 = 0$ при начальном приближении в точке $x_0 = 0$

§ 1.6. Модифицированный метод Ньютона

В ряде случаев бывает полезно воспользоваться формулой *модифицированного метода Ньютона*

$$x_{k+1} = x_k - \frac{f(x_k)}{f'(x_0)}, \quad k = 0, 1, \ldots . \tag{1.6}$$

Здесь значение производной $f'(x_0)$ вычисляется один раз и далее не меняется. Геометрически это означает, что при итерациях проводятся прямые, параллельные касательной в точке $(x_0, f(x_0))$. Этот метод обычно требует большего числа итераций, чем обычный метод Ньютона, но позволяет избежать вычисления на каждом шаге значения производной. Сходимость этого метода также существенно зависит от правильного выбора исходного приближения. Возможны ситуации с зацикливанием итераций.

На рис. 1.10 сходимость итераций по модифицированному методу Ньютона показана на примере решения уравнения $f(x) = x^3 - x - 3 = 0$, $1 \le x \le 3$ с начальным приближением в точке $x_0 = 3$. Приближенное значение корня $x^* \approx 1{,}67$. Применение формулы (1.6) дает:

$$x_1 = 2{,}19, \quad x_2 = 1{,}99, \quad x_3 = 1{,}88, \quad x_4 = 1{,}81, \quad \ldots \quad x_{15} = 1{,}67.$$

§ 1.7. Метод секущих

Пусть значения функции f заданы в двух точках $(x_{k-1}, f(x_{k-1}))$ и $(x_k, f(x_k))$ вблизи ее нуля $(x^*, 0)$. Проведем через эти две точки прямую до ее пересения с осью абсцисс в точке x_{k+1}, ближайшей к x_k. Используя подобие треугольников (рис. 1.11), получаем двухточечную итерационную формулу *метода секущих*:

$$x_{k+1} = x_k - \frac{f(x_k)(x_k - x_{k-1})}{f(x_k) - f(x_{k-1})}, \quad k = 1, 2, \ldots . \tag{1.7}$$

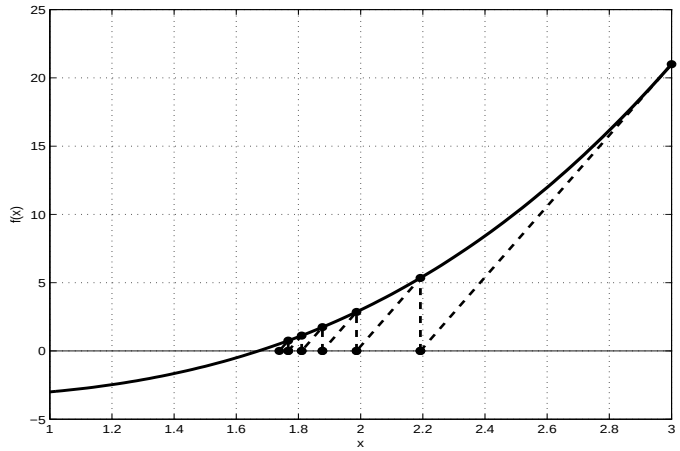


Рис. 1.10. Итерации модифицированного метода Ньютона по поиску корня уравнения $f(x) = x^3 - x - 3 = 0,\ 1 \le x \le 3$

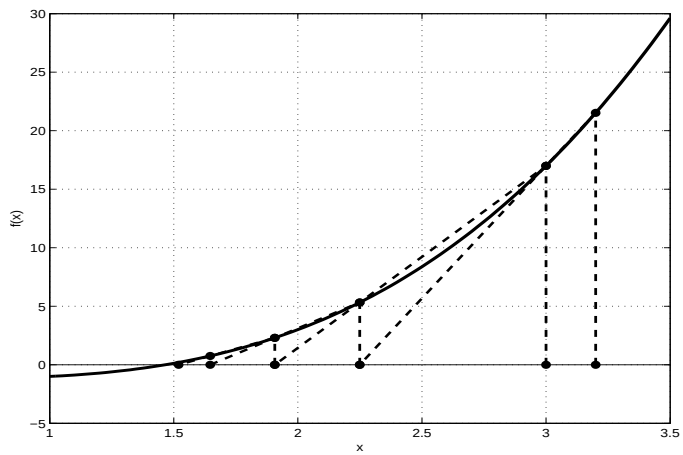


Рис. 1.11. Итерации метода секущих для уравнения $f(x) = x^3 - x^2 - 1 = 0,\ 1 \le x \le 3$ при $x_0 = 3{,}2$ и $x_1 = 3$

Эта формула может быть получена из метода Ньютона заменой значения производной $f'(x_k)$ на разделенную разность $(f(x_k) - f(x_{k-1}))/(x_k - x_{k-1})$. По этой причине метод секущих может рассматриваться как дискретизация метода Ньютона. В свою очередь метод Ньютона может быть получен из метода секущих как предельный частный случай, когда $x_{k-1} \to x_k$.

Рис. 1.11 иллюстрирует применение метода секущих для поиска корня уравнения $f(x) = x^3 - x^2 - 1,\ 1 \le x \le 3$. В качестве начального приближения использовались точки $x_0 = 3{,}2$ и $x_1 = 3$. Приближенное значение корня $x^* \approx 1{,}47$. Применение формулы (1.7) дает:

$$x_2 = 2{,}22, \quad x_3 = 1{,}89, \quad x_4 = 1{,}64, \quad x_5 = 1{,}51, \quad x_6 = 1{,}47.$$

§ 1.8. Скорость сходимости итерационных методов

Следующее определение позволяет сравнивать между собой различные итерационные методы по скорости сходимости.

Определение 1.1. *Пусть дана последовательность $x_0, x_1, \ldots$, сходящаяся к пределу x^*. Тогда, если существует вещественное число $p \geq 1$ такое, что*

$$\lim_{k\to\infty} \frac{|x_{k+1} - x^*|}{|x_k - x^*|^p} = C = const \neq 0,$$

то говорят, что сходимость имеет порядок p. Постоянная C называется асимптотической константой погрешности.

При $p = 1$ сходимость линейная. При $p = 2$ сходимость квадратичная и т. д. В частности, метод деления отрезка пополам имеет линейную скорость сходимости. Покажем, что метод Ньютона имеет квадратичную скорость сходимости.

Пусть $f \in C^2[a, b]$. Обозначим $g(x) = x - f(x)/f'(x)$. Разложим функцию g по формуле Тейлора в окрестности нуля $x = x^*$. Имеем

$$g(x) = g(x^*) + g'(x^*)(x - x^*) + g''(x^* + \theta(x - x^*))\frac{(x - x^*)^2}{2!}, \quad 0 < \theta < 1.$$

Так как $g'(x) = f(x)f''(x)/[f'(x)]^2$, то здесь $g'(x^*) = 0$. Применяя эту формулу для $x = x_i$, получаем

$$x_{i+1} = x^* + \frac{1}{2}g''(x^* + \theta(x_i - x^*))(x_i - x^*)^2,$$

то есть

$$|x_{i+1} - x^*| = |x_i - x^*|^2|g''(x^* + \theta(x_i - x^*))|/2.$$

Теперь, используя ограниченность производных от f, можно показать, что функция g'' ограничена в достаточно малой окрестности данного нуля. Следовательно, если метод Ньютона сходится, то он сходится квадратично.

Выясним скорость сходимости метода секущих. Пусть x^* – корень уравнения (1.1). Тогда по формуле Тейлора

$$0 = f(x^*) = f(x_k) + f'(x_k)(x^* - x_k) + \frac{f''(x_k)}{2}(x^* - x_k)^2 + O((x^* - x_k)^3).$$

Отсюда при обозначении $w_k = x_k - x^*$ имеем

$$f(x_k) = f'(x_k)w_k - \frac{f''(x_k)}{2}w_k^2 + O(w_k^3). \tag{1.8}$$

Аналогично

$$f(x_{k-1}) = f(x_k) + f'(x_k)(x_{k-1} - x_k) + \frac{f''(x_k)}{2}(x_{k-1} - x_k)^2 + O((x_{k-1} - x_k)^3).$$

Поэтому

$$f(x_k) - f(x_{k-1}) = f'(x_k)(w_k - w_{k-1}) - \frac{f''(x_k)}{2}(w_k - w_{k-1})^2 + O((w_k - w_{k-1})^3). \quad (1.9)$$

Учитывая соотношения (1.8) и (1.9), формулу (1.5) можно переписать в виде

$$w_{k+1} = w_k - \frac{(f'(x_k)w_k + f''(x_k)w_k^2/2 + O(w_k^3))(w_k - w_{k-1})}{f'(x_k)(w_k - w_{k-1}) - f''(x_k)(w_k - w_{k-1})^2/2 + O((w_k - w_{k-1})^3)}.$$

В предположении $f'(x^*) \neq 0$ разделим числитель и знаменатель на величину $f'(x_k)(w_k - w_{k-1})$, что дает

$$w_{k+1} = w_k - \frac{w_k - \lambda w_k^2 + O(w_k^3)}{1 - \lambda(w_k - w_{k-1}) + O((w_k - w_{k-1})^2)}, \quad \lambda = \frac{f''(x_k)}{2f'(x_k)}.$$

Так как для малого ε справедливо равенство $1/(1-\varepsilon) = 1+\varepsilon+O(\varepsilon^2)$, то имеем

$$w_{k+1} = w_k - (w_k - \lambda w_k^2 + O(w_k^3))(1 + \lambda(w_k - w_{k-1}) + O((w_k - w_{k-1})^2)$$

или

$$w_{k+1} = \lambda w_k w_{k-1} + O((|w_k| + |w_{k-1}|)^3).$$

Поэтому можно считать, что

$$w_{k+1} \approx \lambda w_k w_{k-1}.$$

Решение этого рекуррентного соотношения будем искать в виде $w_{k+1} = \lambda^\alpha w_k^\beta$. Тогда

$$\lambda^\alpha w_k^\beta = \lambda w_k w_{k-1} \quad \text{или} \quad w_k = \lambda^{\frac{1-\alpha}{\beta-1}} w_{k-1}^{\frac{1}{\beta-1}}.$$

Отсюда:

$$\frac{1-\alpha}{\beta-1} = \alpha, \quad \frac{1}{\beta-1} = \beta,$$

то есть:

$$\alpha\beta = 1, \quad \beta^2 - \beta - 1 = 0.$$

Так как из двух корней $\beta_{1,2} = (1 \pm \sqrt{5})/2$ только положительный соответствует убыванию ошибки, то в методе секущих:

$$x_{k+1} - x^* = \lambda^{1/\beta}(x_k - x^*)^\beta, \quad \beta = (1+\sqrt{5})/2 \approx 1{,}62, \quad 1/\beta \approx 0{,}62.$$

Итак, скорость сходимости метода секущих $p \approx 1{,}62$.

Вместо двухточечной итерационной формулы метода секущих можно применить трехточечную итерационную формулу, известную как *метод парабол*. В этом случае начальное приближение задается в трех точках $(x_j, f(x_j))$, $j = k-2, k-1, k$. В качестве следующего приближения к корню берется ближайшая к x_k точка пересечения этой параболы с осью абсцисс. Этот метод позволяет находить как вещественные так и комплексные корни многочленов. Можно показать (см. [14, с. 147]), что метод парабол имеет скорость сходимости $p \approx 1{,}84$. Использование многоточечных итерационных формул усложняет вычисления, но не позволяет получить скорость сходимости выше квадратичной.

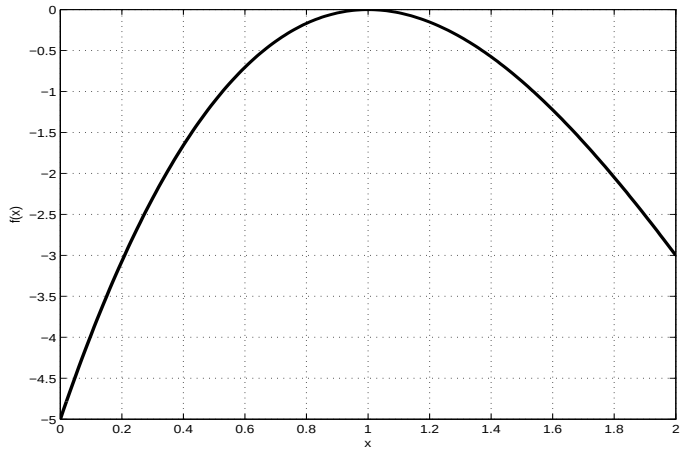


Рис. 1.12. График функции $f(x) = (x-5)(x-1)^2$ с кратным нулем

§ 1.9. Метод Ньютона для поиска кратных корней

Будем считать, что функция f в точке $x = x^*$ имеет нуль кратности m, если

$$f(x^*) = f'(x^*) = \ldots = f^{(m-1)}(x^*) = 0 \quad \text{и} \quad f^{(m)}(x^*) \neq 0.$$

Рассмотренный нами метод Ньютона хорошо работает в случае простых корней, когда $f'(x^*) \neq 0$, т. е. $m = 1$. Однако в случае кратных корней могут возникнуть трудности.

Пример 1.5. Рассмотрим уравнение $f(x) = (x-1)^2(x-5) = 0$. Функция f имеет нуль кратности $m = 2$ в точке $x^* = 1$. График этой функции показан на рис. 1.12.

Итерации начнем из точки $x_0 = 0$. Согласно результатам расчета, приведенным в табл. 1.1, скорость сходимости метода Ньютона оказывается не квадратичной а только линейной

$$\frac{|e_{k+1}|}{|e_k|} = \frac{|\alpha - x_{k+1}|}{|\alpha - x_k|} \approx 0{,}5.$$

Таблица 1.1

k	x_k	$\lvert x_{k+1} - x_k\rvert$	$\lvert x^* - x_k\rvert$	$\lvert e_{k+1}\rvert/\lvert e_k\rvert$
0	0,0000	0,4545	1,0000	0,5454
1	0,4545	0,2573	0,5454	0,5283
2	0,7118	0,1394	0,2882	0,5162
3	0,8512	0,0731	0,1488	0,5088
4	0,9243	0,0375	0,0757	0,5046
5	0,9618	0,0190	0,0382	0,5023
6	0,9808	0,0096	0,0192	0,5012
7	0,9903	0,0048	0,0096	

Выясним как устранить это нежелательное явление. Пусть функция f имеет нуль x^* кратности m. Пользуясь разложением по формуле Тейлора, находим:

$$f(x) = (x - x^*)^m g(x), \quad g(x) = \frac{f^{(m)}(x^*)}{m!} + \frac{f^{(m+1)}(x^*)}{(m+1)!}(x - x^*) + \dots .$$

Дифференцируя это равенство, получаем

$$f'(x) = m(x - x^*)^{m-1} g(x) + (x - x^*)^m g'(x) = (x - x^*)^{m-1}[mg(x) + (x - x^*)g'(x)].$$

Таким образом,

$$h(x) = \frac{f(x)}{f'(x)} = \frac{(x - x^*)g(x)}{mg(x) + (x - x^*)g'(x)} = (x - x^*)\Psi(x).$$

Так как $g(x^*) = f^{(m)}(x^*)/m!$, то

$$\lim_{x \to x^*} h'(x) = \lim_{x \to x^*} [\Psi(x) + (x - x^*)\Psi'(x)] = \frac{1}{m}. \tag{1.10}$$

Следовательно, функция h в точке $x = x^*$ имеет простой нуль.

Записывая теперь метод Ньютона для функции h вблизи нуля $x = x^*$, с учетом равенства (1.10) получаем

$$x_{k+1} = x_k - m\frac{f(x_k)}{f'(x_k)}, \quad k = 0, 1, \dots . \tag{1.11}$$

Здесь m – кратность нуля x^*. Формула (1.11) называется *методом Ньютона для поиска корней кратности* m. В табл. 1.2 приведены результаты расчетов с использованием метода Ньютона (1.11) при $m = 2$ для поиска кратного корня уравнения $f(x) = (x-5)(x-1)^2 = 0$ с точкой начала итераций $x_0 = 0$. Теперь метод Ньютона сходится квадратически и

$$\frac{|e_{k+1}|}{|e_k|^2} = \frac{|x^* - x_{k+1}|}{|x^* - x_k|^2} \approx 0{,}12.$$

Таблица 1.2

k	x_k	$\lvert x_{k+1} - x_k\rvert$	$\lvert x^* - x_k\rvert$	$\lvert e_{k+1}\rvert/\lvert e_k\rvert^2$
0	0,0000	0,9091	1,0000	0,9091
1	0,9091	0,0899	0,0909	0,1209
2	0,9990	0,0010	0,0010	0,1250
3	1,0000	0,0000	0,0001	

§ 1.10. Метод простой итерации

Все рассмотренные нами выше итерационные методы решения нелинейного уравнения (1.1) могут быть истолкованы как частные случаи одного общего метода, известного как *метод простой итерации*.

Перепишем уравнение (1.1) в виде

$$x = \varphi(x),$$

положив, например, $\varphi(x) = x + \psi(x)f(x)$, где $\psi(x)$ – произвольная непрерывная знакопостоянная функция.

Приближения образуем по правилу

$$x_{k+1} = \varphi(x_k), \quad k = 0, 1, \ldots . \tag{1.12}$$

Пусть x^* – корень уравнения (1.1). Очевидно, что $x^* = \varphi(x^*)$, т. е. точное решение является неподвижной точкой отображения φ. Будем считать, что вблизи корня x^* отображение φ является сжимающим, т. е.

$$|x_{k+1} - x_k| = |\varphi(x_k) - \varphi(x_{k-1})| \le q|x_k - x_{k-1}|, \quad q < 1.$$

Если функция φ имеет непрерывную производную, то

$$x_{k+1} - x^* = \varphi(x_k) - \varphi(x^*) = \varphi'(\xi)(x_k - x^*),$$

где ξ находится между x^* и x_k. При выполнении условия

$$|\varphi'(x)| \le q < 1 \tag{1.13}$$

метод сходится и скорость сходимости линейная.

Если x_0 взято достаточно близко к x^*, то последовательность $\{x_k\}$ сходится к x^*, так как существует окрестность точки x^*, в которой выполняется условие (1.13) и для сходимости $\{x_k\}$ нужно только потребовать, чтобы x_0 принадлежало этой окрестности.

Условие (1.13) называют *достаточным условием сходимости метода простых итераций*. Чем меньше будет q, тем быстрее будет сходимость.

При $\varphi'(x) < 0$ приближения попеременно оказываются то слева, то справа от корня. Поэтому для нахождения корня с точностью $\varepsilon > 0$ достаточно воспользоваться грубой оценкой:

$$|x_{k+1} - x_k| < \varepsilon.$$

Если $\varphi'(x) > 0$, то сходимость итераций монотонная, т. е. слева или справа. Так как отображение φ является сжимающим, то вблизи корня итерации сходятся как

геометрическая прогрессия со знаменателем $q = (x_{k+1} - x_k)/(x_k - x_{k-1})$. Чтобы сумма ее последующих членов была не больше ε, должно выполняться условие

$$\left| q\frac{x_{k+1} - x_k}{1-q} \right| = \frac{(x_{k+1} - x_k)^2}{|2x_k - x_{k+1} - x_{k-1}|} < \varepsilon.$$

Так как

$$x_{k+1} - x^* = \varphi'(\xi)(x_k - x^*) = \varphi'(\xi)(x_k - x_{k+1} + x_{k+1} - x^*),$$

то

$$|x_{k+1} - x^*| \le \frac{q|x_{k+1} - x_k|}{1-q} < \varepsilon.$$

Если $\varphi'(x^*) = 0$, то скорость сходимости квадратичная. Действительно, из формулы (1.12), пользуясь разложением по формуле Тейлора, имеем

$$x_{k+1} = \varphi(x^*) + \varphi'(x^*)(x_k - x^*) + \varphi''(\xi)\frac{(x_k - x^*)^2}{2}.$$

Отсюда при условии $\varphi'(x^*) = 0$ получаем

$$x_{k+1} - x^* = \frac{1}{2}\varphi''(\xi)(x_k - x^*)^2,$$

что при ограниченности величины $|\varphi''(\xi)|/2$ означает квадратичную сходимость процесса итераций.

Сходимость метода простой итерации существенно зависит от правильного выбора функции φ.

Рассмотрим уравнение $x^2 = a$ и два способа образования итераций:

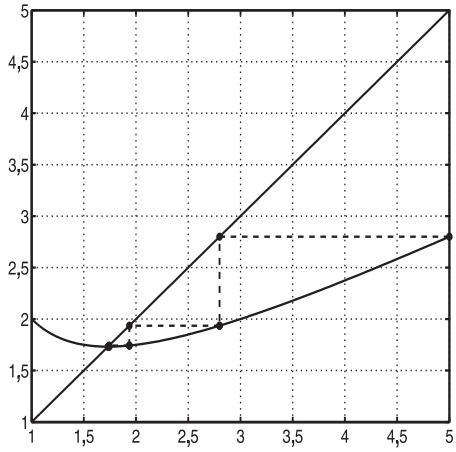
1. $\varphi(x) = a/x$ и $x_{k+1} = a/x_k$;
2. $\varphi(x) = \frac{1}{2}\left(x + \frac{a}{x}\right)$ и $x_{k+1} = \frac{1}{2}\left(x_k + \frac{a}{x_k}\right)$.

В первом случае процесс циклит: $x_{k+2} = x_k$ и сходимости нет. Во втором случае сходимость есть и очень быстрая. Рис. 1.13а иллюстрирует сходимость по второму способу при вычислении $\sqrt{3} \approx 1{,}73$. При $x_0 = 5$ имеем:

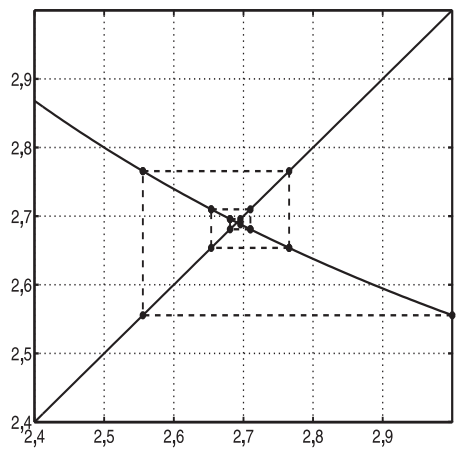
$$x_1 = 2{,}80, \quad x_2 = 1{,}94, \quad x_3 = 1{,}74, \quad x_4 = 1{,}73.$$

На рис. 1.13б показана сходимость типа спирали, получаемая при решении уравнения $x^3 - 2x^2 - 5 = 0$ на отрезке $2{,}5 \le x \le 3$. Перепишем это уравнение в виде $x = 5/x^2 + 2$ и найдем его корень с точностью до 0,01. Применение метода простых итераций (1.12) с $x_0 = 3$ дает:

$$x_1 = 2{,}56, \quad x_2 = 2{,}77, \quad x_3 = 2{,}65, \quad \ldots \quad x_7 = 2{,}68.$$



(а) $0 < \varphi'(x) < 1$



(б) $-1 < \varphi'(x) < 0$

Рис. 1.13. Монотонная (а) и осциллирующая (б) сходимости метода простых итераций

§ 1.11. Метод Чебышева

Получим теперь методы высокого порядка сходимости.

Пусть решается уравнение (1.1). Запишем обратную функцию $x = F(y)$, где $y \in [c, d]$ и $[c, d]$ – область изменения $f(x)$ для $x \in [a, b]$.

Так как $x \equiv F[f(x)]$ и $y = f[F(y)]$, то $x^* = F(0)$. При $y \in [c, d]$ формула Тейлора дает

$$x^* = F(0) = F(y) + \sum_{k=1}^{r} (-1)^k \frac{F^{(k)}(y)}{k!} y^k + R_{r+1},$$

где остаточный член может быть записан в виде

$$R_{r+1} = (-1)^{r+1} \frac{F^{(r+1)}(\eta)}{(r+1)!} y^{r+1},$$

и η лежит между 0 и y, или

$$x^* = x + \sum_{k=1}^{r} (-1)^k \frac{F^{(k)}[f(x)]}{k!} [f(x)]^k 0 + (-1)^{r+1} \frac{F^{(r+1)}(\eta)}{(r+1)!} [f(x)]^{r+1}. \tag{1.14}$$

Для упрощения записи положим:

$$F^{(k)}[f(x)] \equiv a_k(x), \quad \varphi_r(x) \equiv x + \sum_{k=1}^{r} (-1)^k \frac{a_k(x)}{k!} [f(x)]^k. \tag{1.15}$$

Уравнение

$$x = \varphi_r(x)$$

имеет корень $x = x^*$, так как

$$\varphi_r(x^*) = x^* + \sum_{k=1}^{r} (-1)^k \frac{a_k(x^*)}{k!} [f(x^*)]^k = x^*.$$

Положив

$$x_{k+1} = \varphi_r(x_k), \quad k = 0, 1, \ldots, \quad x_0 \in [a, b],$$

получим итерационный метод $(r+1)$-го порядка, так как

$$\varphi_r^{(l)}(x^*) = 0, \quad l = 1, 2, \ldots, r.$$

Функцию φ_r можно найти в явном виде через f и ее производные. Так как $x = F[f(x)]$, то имеем:

$$\begin{aligned}
&F'[f(x)]f'(x) = 1;\\
&F''[f(x)][f'(x)]^2 + F'[f(x)]f''(x) = 0;\\
&F'''[f(x)][f'(x)]^3 + 3F''[f(x)]f'(x)f''(x) + F'[f(x)]f'''(x) = 0,\\
&\qquad \ldots
\end{aligned}$$

или с учетом тождеств (1.15):

$$\begin{aligned}
&a_1(x)f'(x) = 1;\\
&a_2(x)[f'(x)]^2 + a_1(x)f''(x) = 0;\\
&a_3(x)[f'(x)]^3 + 3a_2(x)f'(x)f''(x) + a_1(x)f'''(x) = 0,\\
&\qquad \ldots
\end{aligned}$$

Поэтому можно последовательно найти a_1, a_2, a_3 и т. д. а, следовательно, и φ_r.

При $r = 1$:

$$\varphi_1(x) = x - \frac{f(x)}{f'(x)} \quad \text{и} \quad x_{k+1} = x_k - \frac{f(x_k)}{f'(x_k)}, \quad k = 0, 1, \ldots,$$

то есть мы снова получаем метод Ньютона.

При $r = 2$:

$$\varphi_2(x) = x - \frac{f(x)}{f'(x)} - \frac{f''(x)f^2(x)}{2[f'(x)]^3}$$

и

$$x_{k+1} = x_k - \frac{f(x_k)}{f'(x_k)} - \frac{f''(x_k)f^2(x_k)}{2[f'(x_k)]^3}, \quad k = 0, 1, \ldots, \tag{1.16}$$

– это метод Чебышева третьего порядка сходимости.

Оценка погрешности и скорость сходимости легко находятся из равенства (1.14). Полагая в нем $x = x_k$ и учитывая, что $x_{k+1} = \varphi_r(x_k)$, получим

$$x^* - x_{k+1} = (-1)^{r+1}\frac{F^{(r+1)}[f(\xi)]}{(r+1)!}[f(x_k)]^{r+1}, \tag{1.17}$$

где ξ лежит между x^* и x_k. Если положить

$$L = \max_{x\in[a,b]} |f'(x)|, \quad M_{r+1} = \max_{x\in[a,b]} |F^{(r+1)}[f(x)]|$$

и учесть, что

$$|f(x_k)| = |f(x_k) - f(x^*)| = |f'(\eta)||x_k - x^*| \le L|x_k - x^*|,$$

то из (1.17) имеем

$$|x^* - x_{k+1}| \le \frac{M_{r+1}L^{r+1}}{(r+1)!}|x_k - x^*|^{r+1} = q|x_k - x^*|^{r+1}, \quad q = \frac{M_{r+1}L^{r+1}}{(r+1)!}.$$

Таким образом, имеем сходимость с порядком $(r+1)$.

Пример 1.6. Пользуясь методами Ньютона и Чебышева, найдем три верных знака корня уравнения $e^x - 1 = 0$. Согласно формулам (1.5) и (1.16) при $x_0 = 1$ имеем:

$$\begin{aligned} x_1 &= 0{,}3679; \quad x_2 = 0{,}0600; \quad x_3 = 0{,}0012; \quad x_4 = 0{,}0000; \\ x_1 &= 0{,}1681; \quad x_2 = 0{,}0014; \quad x_3 = 0{,}0000; \end{aligned}$$

т. е. метод Чебышева сходится быстрее, чем метод Ньютона.

§ 1.12. Метод Эйткена построения итераций высших порядков

Этот способ позволяет из данной итерации или из двух итераций одного и того же порядка получать итерации более высокого порядка.

Пусть имеются итерации:

$$x_k^{(1)} = \varphi_1(x_{k-1}^{(1)}), \quad x_k^{(2)} = \varphi_2(x_{k-1}^{(2)}), \quad k = 1, 2, \ldots$$

порядка r, сходящиеся к $x = x^*$. Используя функции φ_1 и φ_2, строим функцию Φ:

$$\Phi(x) = \frac{x\varphi_1[\varphi_2(x)] - \varphi_1(x)\varphi_2(x)}{x - \varphi_1(x) - \varphi_2(x) + \varphi_1[\varphi_2(x)]}.$$

Тогда итерации (см. [3, т. 1, с. 480]):

$$x_k = \Phi(x_{k-1}), \quad k = 1, 2, \ldots$$

имеют порядок выше r, если только

$$[\varphi_1'(x^*) - 1][\varphi_2'(x^*) - 1] \ne 0.$$

В частности, можно положить $\varphi(x) = \varphi_1(x) = \varphi_2(x)$. Тогда

$$\Phi(x) = \frac{x\varphi[\varphi(x)] - \varphi^2(x)}{x - 2\varphi(x) + \varphi[\varphi(x)]}.$$

При реализации итераций поступаем следующим образом:

$$x_1 = \varphi(x_0), \quad x_2 = \varphi(x_1), \quad x_3 = \frac{x_0 x_2 - x_1^2}{x_0 - 2x_1 + x_2} = x_0 - \frac{(\Delta x_0)^2}{\Delta^2 x_0},$$

где $\Delta x_0 = x_1 - x_0$.

Аналогично находим:

$$x_4 = \varphi(x_3), \quad x_5 = \varphi(x_4), \quad x_6 = \frac{x_3 x_5 - x_4^2}{x_3 - 2x_4 + x_5} = x_3 - \frac{(\Delta x_3)^2}{\Delta^2 x_3},$$

и т. д.

Итак, получаем итерационный процесс:

$$x_{3k+1} = \varphi(x_{3k}), \quad x_{3k+2} = \varphi(x_{3k+1}), \quad x_{3k+3} = x_{3k} - \frac{(\Delta x_{3k})^2}{\Delta^2 x_{3k}}, \quad k = 0, 1, \ldots,$$

где

$$\Delta x_{3k} = x_{3k+1} - x_{3k}, \quad \Delta^2 x_{3k} = x_{3k+2} - 2x_{3k+1} + x_{3k}.$$

Точно так же, как по φ строилась итерация Φ более высокого порядка, можно по Φ построить итерацию еще более высокого порядка и т. д.

§ 1.13. Решение систем нелинейных уравнений

Рассмотрим систему n нелинейных уравнений:

$$\begin{cases} f_1(x_1, x_2, \ldots, x_n) = 0; \\ f_2(x_1, x_2, \ldots, x_n) = 0; \\ \qquad\qquad \ldots \\ f_n(x_1, x_2, \ldots, x_n) = 0. \end{cases}$$

Пользуясь разложением по формуле Тейлора, имеем

$$f_i(x_1 + \Delta x_1, \ldots, x_n + \Delta x_n) =$$
$$= f_i(x_1, \ldots, x_n) + \Delta x_1 \frac{\partial f_i}{\partial x_1} + \ldots + \Delta x_n \frac{\partial f_i}{\partial x_n} + \ldots = 0, \quad i = 1, \ldots, n.$$

Отсюда, ограничиваясь слагаемыми первого порядка малости, получаем

$$\begin{pmatrix} \frac{\partial f_1}{\partial x_1} & \frac{\partial f_1}{\partial x_2} & \ldots & \frac{\partial f_1}{\partial x_n} \\ \frac{\partial f_2}{\partial x_1} & \frac{\partial f_2}{\partial x_2} & \ldots & \frac{\partial f_2}{\partial x_n} \\ & & \ldots & \\ \frac{\partial f_n}{\partial x_1} & \frac{\partial f_n}{\partial x_2} & \ldots & \frac{\partial f_n}{\partial x_n} \end{pmatrix} \begin{pmatrix} \Delta x_1 \\ \Delta x_2 \\ \vdots \\ \Delta x_n \end{pmatrix} = \begin{pmatrix} -f_1 \\ -f_2 \\ \vdots \\ -f_n \end{pmatrix}. \tag{1.18}$$

Задавая начальное приближение $x^{(0)} = (x_1^{(0)}, x_2^{(0)}, \ldots, x_n^{(0)})$ и решая эту систему, находим приращения $\Delta x_i^{(0)}$, $i = 1, 2, \ldots, n$, и, как следствие, с помощью пересчета

$$x^{(1)} = x^{(0)} + \Delta x^{(0)}$$

образуем новые приближенные значения неизвестных. Повторное выполнение этих вычислений дает нам метод Ньютона для системы n нелинейных уравнений. Матрица системы (1.18) называется матрицей Якоби и обозначается $J(x_1, \ldots, x_n)$. Если начальное приближение выбрано достаточно близко к решению и матрица Якоби не вырождена, то метод Ньютона должен сходиться квадратично.

Проиллюстрируем этот метод на примере решения системы двух нелинейных уравнений:

$$\begin{cases} f_1(x_1, x_2) = 0; \\ f_2(x_1, x_2) = 0. \end{cases}$$

Здесь метод Ньютона принимает вид:

$$J \begin{pmatrix} x_1^{(k+1)} - x_1^{(k)} \\ x_2^{(k+1)} - x_2^{(k)} \end{pmatrix} = \begin{pmatrix} -f_1(x_1, x_2) \\ -f_2(x_1, x_2) \end{pmatrix}^{(k)}, \quad J = \begin{pmatrix} \frac{\partial f_1}{\partial x_1} & \frac{\partial f_1}{\partial x_2} \\ \frac{\partial f_2}{\partial x_1} & \frac{\partial f_2}{\partial x_2} \end{pmatrix}, \quad k = 0, 1, \ldots$$

или

$$\begin{pmatrix} x_1 \\ x_2 \end{pmatrix}^{(k+1)} = \begin{pmatrix} x_1 \\ x_2 \end{pmatrix}^{(k)} - J^{-1} \begin{pmatrix} f_1(x_1, x_2) \\ f_2(x_1, x_2) \end{pmatrix}^{(k)}, \quad k = 0, 1, \ldots,$$

где

$$J^{-1} = \frac{1}{|J|} \begin{pmatrix} \frac{\partial f_2}{\partial x_2} & -\frac{\partial f_1}{\partial x_2} \\ -\frac{\partial f_2}{\partial x_1} & \frac{\partial f_1}{\partial x_1} \end{pmatrix}, \quad |J| = \frac{\partial f_1}{\partial x_1}\frac{\partial f_2}{\partial x_2} - \frac{\partial f_1}{\partial x_2}\frac{\partial f_2}{\partial x_1}.$$

В покомпонентной форме имеем

$$\begin{cases} x_1^{(k+1)} = x_1^{(k)} - \Big[\dfrac{f_1}{|J|}\dfrac{\partial f_2}{\partial x_2} - \dfrac{f_2}{|J|}\dfrac{\partial f_1}{\partial x_2}\Big]^{(k)}; \\ x_2^{(k+1)} = x_2^{(k)} - \Big[\dfrac{f_2}{|J|}\dfrac{\partial f_1}{\partial x_1} - \dfrac{f_1}{|J|}\dfrac{\partial f_2}{\partial x_1}\Big]^{(k)}, \quad k = 0, 1, \ldots. \end{cases} \tag{1.19}$$

Пример 1.7. Пусть требуется решить систему

$$\begin{cases} f_1(x, y) = x^2 + y^2 - 1 = 0, \\ f_2(x, y) = 4xy - 1 = 0 \end{cases} \tag{1.20}$$

и найти ее корни с точностью до 0,01.

Итерационные формулы (1.19) принимают вид:

$$\begin{cases} x_{k+1} = x_k - \dfrac{0{,}5}{x_k^2 - y_k^2}\Big[x_k f_1(x_k, y_k) - 2y_k f_2(x_k, y_k)\Big], \\ y_{k+1} = y_k - \dfrac{0{,}5}{x_k^2 - y_k^2}\Big[-y_k f_1(x_k, y_k) + 2x_k f_2(x_k, y_k)\Big], \quad k = 0, 1, \ldots. \end{cases} \tag{1.21}$$

Условие $\max(|x_{k+1} - x_k|, |y_{k+1} - y_k|) < 0{,}01$ используем для прекращения итераций.

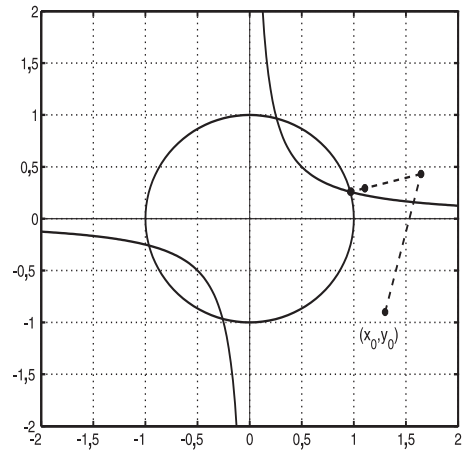


Рис. 1.14. Итерации по методу Ньютона для системы уравнений (1.20)

Выбирая начальное приближение в точке $(x_0, y_0) = (1{,}3; -0{,}9)$, по формулам (1.21) последовательно получаем:

$$\begin{aligned} x_1 &= 1{,}6443; \quad x_2 = 1{,}1059; \quad x_3 = 0{,}9748; \quad x_4 = 0{,}9660; \\ y_1 &= 0{,}4307; \quad y_2 = 0{,}2931; \quad y_3 = 0{,}2608; \quad y_4 = 0{,}2588. \end{aligned} \tag{1.22}$$

Остальные корни системы (1.20) находятся из соображений симметрии. Рис. 1.14 иллюстрирует сходимость метода Ньютона в этом примере.

§ 1.14. Задачи

1.1. Рассмотрите метод хорд как частный случай метода простых итераций. Покажите, что метод хорд имеет линейную скорость сходимости.

1.2. Какую скорость сходимости имеет модифицированный метод Ньютона?

1.3. Пусть уравнение $f(x) = 0$ на отрезке $[a, b]$ имеет единственный корень x^* кратности $m > 1$ и f – дважды дифференцируемая функция. Покажите, что тогда метод Ньютона сходится линейно как геометрическая прогрессия со знаменателем $(m-1)/m$.

1.4. Получите расчетные формулы для метода парабол и докажите, что этот метод сходится со скоростью $p \approx 1{,}84$.

1.5. Уравнение $tg(x) = 1 + x$ требуется решить методом простой итерации. Какое из приводимых выражений подходит для этой цели:

а) $x = -1 + tg(x)$; б) $x = arctg(1 + x)$?

1.6. Пусть уравнение $f(x) = 0$ имеет на отрезке $[a, b]$ единственный корень x^* и $f'(x) \cdot f''(x) > 0 (< 0)$ для всех $x \in [a, b]$. Покажите, что тогда при $x_0 = b$ $(= a)$ итерационная последовательность, построенная по методу Ньютона, монотонно убывает (возрастает) и сходится к корню x^*.

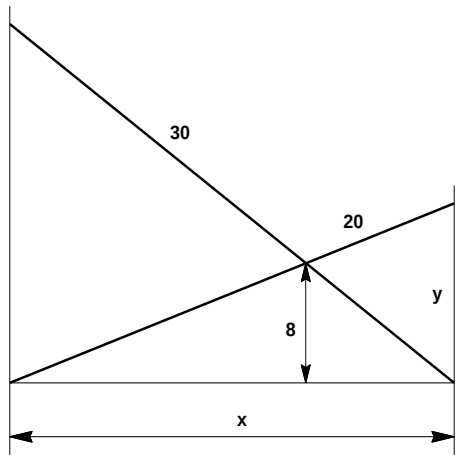


Рис. 1.15. Задача о двух лестницах, приводящая к нелинейному уравнению.

1.7. Выясните условия сходимости метода Ньютона для уравнения $x^2 - 9 = 0$.

1.8. Для решения уравнения $f(x) = 0$ рассмотрим *метод релаксации*

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

где $\tau = const$ – параметр релаксации. Пусть функция f имеет ограниченную и знакопостоянную производную, т. е. $0 < m < |f'(x)| < M$. Найдите оптимальное значение параметра релаксации τ, при котором метод сходится быстрее всего. Покажите, что метод релаксации имеет линейную скорость сходимости.

1.9. (Старинная задача). Две лестницы, одна в 20 м, другая в 30 м длиной, поставлены поперек улицы, как показано на рис. 1.15, и опираются своими концами на противостоящие дома. Определить ширину улицы, если точка пересечения лестниц находится на высоте 8 м над землей.

Покажите, что эта задача сводится к решению уравнения

$$y^4 - 16y^3 + 500y^2 - 8000y + 32000 = 0$$

и тогда $x = \sqrt{400 - y^2}$.

1.10. Напишите расчетные формулы метода Ньютона для систем:

а) $\begin{cases} y = \sin(x + 1); \\ x^2 + y^2 = 1, \ x > 0, \ y > 0; \end{cases}$ б) $\begin{cases} y = (x - 0{,}5)^2 + 0{,}5; \\ x^2 + y^2 = 1, \ x > 0, \ y > 0. \end{cases}$

1.11. Обобщение метода простой итерации на случай системы двух уравнений

$$\begin{cases} x = \varphi_1(x, y); \\ y = \varphi_2(x, y). \end{cases}$$

дается формулами:

$$\begin{cases} x_{k+1} = \varphi_1(x_k, y_k); \\ y_{k+1} = \varphi_2(x_k, y_k). \end{cases}$$

Покажите, что достаточные условия сходимости этого метода записываются в виде:

$$\left|\frac{\partial \varphi_1}{\partial x}\right| + \left|\frac{\partial \varphi_2}{\partial x}\right| < 1 \quad \text{и} \quad \left|\frac{\partial \varphi_1}{\partial y}\right| + \left|\frac{\partial \varphi_2}{\partial y}\right| < 1.$$

Исследуйте верхнюю границу величины $|x^* - x_k| + |y^* - y_k|$, где (x^*, y^*) – точное решение системы.

Глава 2
Интерполяция

В этой главе рассмотрены вопросы интерполяции классическими многочленами Лагранжа и Ньютона, их кусочными аналогами, называемыми лагранжевыми сплайнами, и наиболее употребительными на практике кубическими сплайнами. Подробно изучаются такие распространенные конструкции, как кусочно-линейная интерполяция и кусочно-кубические многочлены Лагранжа. Показано, как с помощью простых приемов можно получить гладкие аналоги лагранжевых сплайнов, известные как локально-аппроксимационные сплайны. Хотя они и не обладают свойством интерполяции, но дают практически такую же точность приближения, что и соответствующие интерполяционные сплайны. Центральное место в главе занимают алгоритмы построения интерполяционных кубических сплайнов, играющих важнейшую роль в практических методах аппроксимации сплайнами. Достаточные для многих приложений свойства гладкости таких сплайнов сочетаются с простотой их компьютерной реализации и высокой точностью получаемых результатов.

§ 2.1. Постановка задачи интерполяции

Пусть функция f определена на отрезке $[a, b]$ и задана таблицей чисел (x_i, f_i), $i = 0, 1, \ldots, N$, где $f_i = f(x_i)$, а точки x_i образуют упорядоченную последовательность $a = x_0 < x_1 < \ldots < x_N = b$.

Типичная *задача интерполяции* состоит в поиске легко вычислимой функции P_N из заданного класса функций такой, что график P_N проходит через исходные данные, т. е. $P_N(x_i) = f_i$, $i = 0, 1, \ldots, N$, где точки x_i называются *узлами интерполяции*.

Исторически традиционным и наиболее простым способом решения задачи интерполяции является построение *интерполяционного многочлена* P_N степени N, образованного линейной комбинацией $N + 1$ мономов $1, x, \ldots, x^N$.

Условия интерполяции:

$$P_N(x_i) = \sum_{j=0}^{N} a_j x_i^j = f_i, \quad i = 0, 1, \ldots, N \tag{2.1}$$

равносильны системе линейных алгебраических уравнений

$$\begin{pmatrix} 1 & x_0 & \dots & x_0^N \\ 1 & x_1 & \dots & x_1^N \\ \vdots & \vdots & \ddots & \vdots \\ 1 & x_N & \dots & x_N^N \end{pmatrix} \begin{pmatrix} a_0 \\ a_1 \\ \vdots \\ a_N \end{pmatrix} = \begin{pmatrix} f_0 \\ f_1 \\ \vdots \\ f_N \end{pmatrix}. \tag{2.2}$$

Матрица системы (2.2) называется матрицей Вандермонда, а ее определитель – определителем Вандермонда. Так как $x_i \neq x_j$ при $i \neq j$, то определитель Вандермонда $D = \prod_{0 \leq i < j \leq N} (x_j - x_i)$ отличен от нуля и, следовательно, система (2.2) имеет единственное решение. Это доказывает существование и единственность интерполяционного многочлена степени $\leq N$. Тем не менее прямое решение системы (2.2), вообще говоря, нецелесообразно, так как обычно матрица системы (2.2) (с «почти» линейно зависимыми строками) плохообусловлена. Для вычисления значений интерполяционного многочлена гораздо более эффективным оказывается использование интерполяционных формул Лагранжа или Ньютона, позволяющих выписать решение системы (2.2) в явном виде.

§ 2.2. Интерполяционные многочлены Лагранжа

Введем в рассмотрение *интерполяционный многочлен Лагранжа*

$$L_N(x) = \sum_{j=0}^{N} f_j l_j(x), \tag{2.3}$$

где *фундаментальные многочлены Лагранжа* l_j имеют вид:

$$l_j(x) = \frac{(x - x_0) \dots (x - x_{j-1})(x - x_{j+1}) \dots (x - x_N)}{(x_j - x_0) \dots (x_j - x_{j-1})(x_j - x_{j+1}) \dots (x_j - x_N)}, \quad j = 0, 1, \dots, N$$

или в более компактной форме записи:

$$\begin{aligned} l_j(x) &= \frac{\omega_N(x)}{(x - x_j)\omega_N'(x_j)}, \quad j = 0, 1, \dots, N, \\ \omega_N(x) &= (x - x_0)(x - x_1) \dots (x - x_N) \end{aligned} \tag{2.4}$$

и обладают свойством

$$l_j(x_i) = \begin{cases} 1, & \text{если } i = j, \\ 0 & \text{в противном случае.} \end{cases}$$

График фундаментального многочлена l_j при целочисленной сетке узлов интерполяции $x_i = i$, $i = 0, 1, \dots, 10$ ($N = 10$) изображен на рис. 2.1.

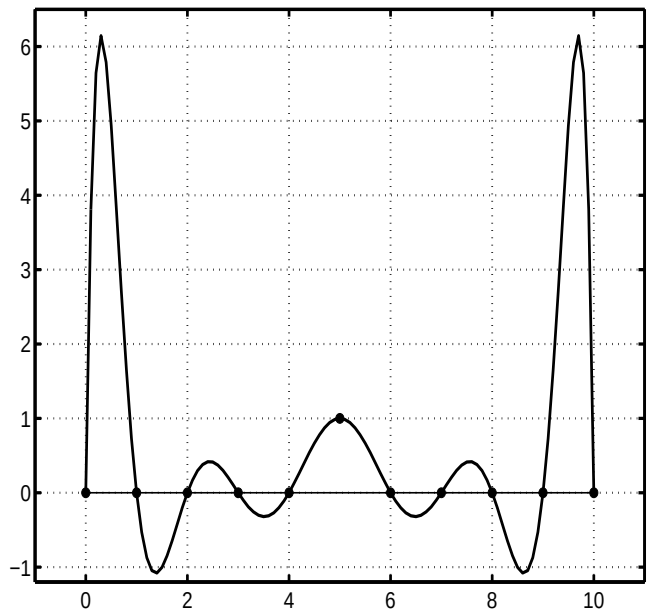


Рис. 2.1. Фундаментальный многочлен Лагранжа с целочисленными узлами.

Непосредственной проверкой убеждаемся, что

$$L_N(x_i) = f_i, \quad i = 0, 1, \ldots, N.$$

В силу формул (2.3) и (2.4) число арифметических операций, требующихся для вычисления значения многочлена Лагранжа, пропорционально N^2.

Лемма 2.1. *Многочлен Лагранжа степени N точен на многочленах степени не выше N, т. е. для всякого многочлена P_k степени $k \leq N$ справедливо тождество*

$$P_k(x) = \sum_{j=0}^{N} P_k(x_j) l_j(x), \quad 0 \leq k \leq N.$$

Доказательство. Достаточно проверить справедливость утверждения на мономах, т. е. доказать тождества

$$x^k \equiv \sum_{j=0}^{N} x_j^k l_j(x), \quad k = 0, 1, \ldots, N.$$

Так как многочлен степени N

$$F_{k,N}(x) = x^k - \sum_{j=0}^{N} x_j^k l_j(x), \quad 0 \leq k \leq N,$$

имеет $N + 1$ нулей: $F_{k,N}(x_i) = 0$, $i = 0, \ldots, N$, то по основной теореме алгебры он тождественно равен нулю. Лемма доказана.

Оценим теперь точность интерполяции многочленом Лагранжа.

Теорема 2.1. *Если функция $f \in C^{N+1}[a,b]$, то для всякого $x \in [a,b]$ найдется точка $\xi_x \in (a,b)$ такая, что*

$$f(x) - L_N(x) = \frac{1}{(N+1)!} f^{(N+1)}(\xi_x)\omega_N(x). \tag{2.5}$$

Доказательство. Если x совпадает с одной из точек интерполяции, то равенство (2.5) очевидно. Предположим поэтому, что x отлично от узла интерполяции и рассмотрим функцию

$$\Phi(x) = f(x) - L_N(x) - C\omega_N(x),$$

где постоянная C берется таким образом, чтобы выполнялось равенство $\Phi(x) = 0$, т. е. $C = [f(x) - L_N(x)]\omega_N^{-1}(x)$.

Так как функция Φ имеет $N+2$ нуля, то, последовательно применяя теорему Ролля [30], получаем, что функция $\Phi^{(N+1)}$ имеет, по крайней мере, один нуль, скажем ξ_x, на (a,b). Тогда

$$\begin{aligned} \Phi^{(N+1)}(\xi_x) &= f^{(N+1)}(\xi_x) - C(N+1)! = \\ &= f^{(N+1)}(\xi_x) - (N+1)!\frac{f(x) - L_N(x)}{\omega_N(x)} = 0, \end{aligned}$$

откуда следует равенство (2.5). Теорема доказана.

Пример 2.1. Пусть функция $f(x) = \sin x$ интерполируется многочленом Лагранжа в 10 точках на отрезке $[0,1]$. Оценить ошибку приближения.

Решение. Применим теорему 2.1. Очевидно, что $|f^{(9)}(\xi_x)| \le 1$ и так как $|(x - x_0)\dots(x - x_9)| \le 1$, то согласно равенству (2.5),

$$|\sin x - L_{10}(x)| \le \frac{1}{10!} < 2{,}8 \cdot 10^{-7}.$$

Пусть $L_{k,0}$ и $L_{k,1}$ – интерполяционные многочлены Лагранжа для данных (x_i, f_i) при $i = 0, 1, \dots, k-1$ и $i = 1, 2, \dots, k$ соответственно.

Лемма 2.2. *Справедлива рекуррентная формула Эйткена*

$$L_{k,0}(x) = \frac{x_k - x}{x_k - x_0} L_{k-1,0}(x) + \frac{x - x_0}{x_k - x_0} L_{k-1,1}(x), \quad k = 1, 2, \dots, N. \tag{2.6}$$

Доказательство Многочлен в правой части формулы (2.6) имеет степень $\le k$ и интерполирует данные (x_i, f_i), $i = 0, 1, \dots, k$. Так как разность двух интерполяционных многочленов степени k имела бы $k+1$ нулей и, следовательно, была бы тождественно равна нулю, то такой интерполяционный многочлен единствен и он совпадает с интерполяционным многочленом Лагранжа $L_k \equiv L_{k,0}$. Лемма доказана.

§ 2.3. Интерполяционные многочлены Ньютона

Рассмотрим рекуррентное соотношение для многочленов Лагранжа другого вида:

$$L_k(x) = L_{k-1}(x) + c_k(x - x_0)\dots(x - x_{k-1}), \quad k = 1, 2, \dots, N.$$

В силу условия интерполяции $L_k(x_k) = f_k$, здесь

$$c_k = \frac{f_k - L_k(x_k)}{(x_k - x_0)\dots(x_k - x_{k-1})} = f[x_0, \dots, x_k].$$

Данное обозначение обычно называется *разделенной разностью* порядка k. В частности, при $k = 0$ полагается $c_0 = f[x_0] \equiv f_0$. Таким образом,

$$L_k(x) = L_{k-1}(x) + f[x_0, \dots, x_k](x - x_0)\dots(x - x_{k-1}). \tag{2.7}$$

Так как, согласно формулам (2.3) и (2.4),

$$L_k(x) = \sum_{j=0}^{k} f_j \frac{\omega_k(x)}{(x - x_j)\omega_k'(x_j)}, \tag{2.8}$$

то, сравнивая коэффициенты при x^k в формулах (2.7) и (2.8), приходим к соотношению

$$f[x_0, \dots, x_k] = \sum_{j=0}^{k} \frac{f_j}{\omega_k'(x_j)}. \tag{2.9}$$

Пусть $(i_0, i_1, \dots, i_k)$ – некоторая перестановка целых чисел $(0, 1, \dots, k)$. Так как

$$f[x_{i_0}, \dots, x_{i_k}] = \sum_{j=0}^{k} \frac{f_{i_j}}{\omega_k'(x_{i_j})} = \sum_{j=0}^{k} \frac{f_j}{\omega_k'(x_j)} = f[x_0, \dots, x_k],$$

то разделенные разности инвариантны относительно перестановок их аргументов.

Формулу Эйткена (2.6) можно переписать в виде

$$L_{k+1}(x) = L_k(x) + \frac{x - x_0}{x_k - x_0}[L_{k,1}(x) - L_{k,0}(x)]. \tag{2.10}$$

Так как:

$$\begin{aligned} L_{k,0}(x) &= f_1 + f[x_1, x_2](x - x_1) + \dots \\ &\quad \dots + f[x_1, \dots, x_{k-1}, x_0](x - x_1)\dots(x - x_{k-1}), \\ L_{k,1}(x) &= f_1 + f[x_1, x_2](x - x_1) + \dots \\ &\quad \dots + f[x_1, \dots, x_{k-1}, x_k](x - x_1)\dots(x - x_{k-1}), \end{aligned}$$

то, пользуясь свойством инвариантности разделенных разностей относительно перестановок их аргументов, получаем

$$L_{k,1}(x) - L_{k,0}(x) = (f[x_1, \dots, x_k] - f[x_0, \dots, x_{k-1}])(x - x_1)\dots(x - x_{k-1}).$$

Подставляя это выражение в формулу (2.10) и сравнивая полученный результат с равенством (2.7), приходим к рекуррентной формуле

$$f[x_0, \ldots, x_k] = \frac{f[x_1, \ldots, x_k] - f[x_0, \ldots, x_{k-1}]}{x_k - x_0}.$$

Суммируя равенства (2.7) по k от 1 до N, получаем *интерполяционную формулу Ньютона*:

$$L_N(x) = L_1(x) + \sum_{k=1}^{N} f[x_0, \ldots, x_k]\omega_{k-1}(x)$$

или

$$L_N(x) = c_0 + c_1(x - x_0) + \ldots + c_N(x - x_0)\ldots(x - x_{N-1}), \tag{2.11}$$

где

$$\begin{aligned} c_0 &= f[x_0] \equiv f_0, \\ c_k &= \frac{f[x_1, \ldots, x_k] - f[x_0, \ldots, x_{k-1}]}{x_k - x_0}, \quad k = 1, 2, \ldots, N. \end{aligned} \tag{2.12}$$

Рассмотрим многочлен Ньютона, который интерполирует функцию f в точках $x_0, \ldots, x_N, t$, где $t \neq x_i$, $i = 0, 1, \ldots, N$. Тогда, согласно формуле (2.11),

$$L_{N+1}(x) = L_N(x) + f[x_0, \ldots, x_N, t]\omega_N(x). \tag{2.13}$$

Так как $L_{N+1}(t) = f(t)$, то, полагая в равенстве (2.13) $x = t$, получаем

$$f(t) - L_N(t) = f[x_0, \ldots, x_N, t]\omega_N(t). \tag{2.14}$$

Сравнивая эту формулу с равенством (2.5), делаем вывод

$$f[x_0, \ldots, x_N, x] = \frac{1}{(N+1)!} f^{(N+1)}(\xi_x).$$

Если положить $x = x_{N+1}$ и $N = n - 1$, то эту формулу можно переписать в симметричном виде:

$$f[x_0, \ldots, x_n] = \frac{f^{(n)}(\xi)}{n!} \tag{2.15}$$

для некоторого $\xi \in [x_0, x_n]$. Отметим, что если f – многочлен степени N вида (2.11), то:

$$f[x_0, \ldots, x_n, x] = \begin{cases} \text{многочлен степени } N - n - 1, & \text{если } n < N - 1, \\ c_N, & \text{если } n = N - 1, \\ 0, & \text{если } n > N - 1. \end{cases}$$

Доказательство этого равенства легко может быть получено по индукции.

При практических вычислениях использование интерполяционных многочленов Ньютона значительно удобнее и экономичнее, чем аналогичных многочленов Лагранжа. При появлении дополнительной точки интерполяции в выражение для интерполяционного многочлена Ньютона надо добавить лишь одно слагаемое в то время как многочлен Лагранжа надо пересчитывать полностью.

§ 2.4. Обобщенная схема Горнера

Формально для нахождения значения l-й производной интерполяционного многочлена Ньютона $L_N^{(l)}$, $0 \le l \le N$ при $x = z$, где z – некоторое заданное число, можно рассмотреть в равенстве (2.11) замену $x = y + z$, где y – новая переменная. Подставляя $x = y + z$ в равенство (2.11) и приводя подобные, получаем

$$L_N(y + z) = A_0 + A_1 y + \ldots + A_N y^N,$$

где $A_l = L_N^{(l)}(z)/l!$, $l = 0, 1, \ldots, N$.

Обратной подстановкой $y = x - z$ находим

$$L_N(x) = A_0 + A_1(x - z) + \ldots + A_N(x - z)^N. \tag{2.16}$$

Нас интересует, однако, наиболее экономичный метод вычисления значений интерполяционного многочлена Ньютона и его производных, известный как *метод расстановки скобок* или *схема Горнера*.

В формуле (2.11) введем переобозначение $a_{i,0} = c_i$, $i = 0, 1, \ldots, N$ и перепишем многочлен L_N в виде

$$L_N(x) \equiv P_{N,0}(x) = a_{0,0} + \sum_{i=1}^{N} a_{i,0}\, \omega_{i-1}(x). \tag{2.17}$$

Путем расстановки скобок представление (2.17) преобразуем к виду

$$\begin{aligned} P_{N,0}(x) &= a_{0,0} + (x - x_0)(a_{1,0} + \ldots \\ &\quad \ldots + (x - x_{N-2})(a_{N-1,0} + (x - x_{N-1})a_{N,0}) \ldots). \end{aligned} \tag{2.18}$$

Для нахождения значения многочлена $P_{N,0}$ в точке $x = z$ образуем последовательность чисел:

$$\begin{aligned} a_{N,1} &= a_{N,0}, \\ a_{i,1} &= a_{i,0} + (z - x_i)a_{i+1,1}, \quad i = N - 1, \ldots, 0. \end{aligned} \tag{2.19}$$

Из формул (2.18) и (2.19) следует, что $L_N(z) \equiv P_{N,0}(z) = a_{0,1}$. Для нахождения значения интерполяционного многочлена требуется выполнить только N умножений и N сложений.

Для нахождения значений производных многочлена Ньютона рассмотрим многочлены:

$$\begin{aligned} P_{N,j}(x) &= a_{j,j} + \sum_{i=j+1}^{N} a_{i,j}\omega_{i-j-1}(x) = a_{j,j} + (x - x_0)(a_{j+1,j} + \ldots \\ &\quad \ldots + (x - x_{N-j-2})(a_{N-1,j} + (x - x_{N-j-1})a_{N,j}) \ldots), \\ &\quad j = 0, 1, \ldots, N. \end{aligned} \tag{2.20}$$

Положим:

$$\begin{aligned} a_{N,j+1} &= a_{N,j}, \\ a_{i,j+1} &= a_{i,j} + (z - x_{i-j})a_{i+1,j+1}, \quad i = N-1, \ldots, j. \end{aligned} \tag{2.21}$$

Из формул (2.20) и (2.21) следует, что $P_{N,j}(z) = a_{j,j+1}$ $(0 \le j \le N)$ и для нахождения значения многочлена $P_{N,j}$ в точке $x = z$ требуется выполнить только $N - j$ умножений и $N - j$ сложений.

Лемма 2.3. *Положим $P_{N,N} \equiv 0$. Справедливы равенства*

$$P_{N,j}(x) = P_{N,j}(z) + (x - z)P_{N,j+1}(x), \quad j = 0, 1, \ldots, N. \tag{2.22}$$

Доказательство. Обозначим $i = N - j$. При $i = 0$ равенство (2.22) очевидно, так как согласно равенству (2.20) имеем $P_{N,N}(x) = P_{N,N}(z) = a_{N,N}$. Пусть равенство (2.22) выполняется для $i = 0, 1, \ldots, N - j'$, где $j < j' \le N$. Покажем, что оно справедливо и при $i = N - j$. Пользуясь формулами (2.21), получаем:

$$\begin{aligned} & P_{N,j}(z) + (x - z)P_{N,j+1}(x) = \\ = \; & a_{j,j+1} + (x - z)\Big(a_{j+1,j+1} + \sum_{i=j+2}^{N} a_{i,j+1}\omega_{i-j-2}(x)\Big) = \\ = \; & a_{j,j+1} + (x - x_0 + x_0 - z)a_{j+1,j+1} + (x - z)\sum_{i=j+2}^{N} a_{i,j+1}\omega_{i-j-2}(x) = \\ = \; & a_{j,j} + (x - x_0)a_{j+1,j+1} + (x - z)\sum_{i=j+2}^{N} a_{i,j+1}\omega_{i-j-2}(x) = \\ = \; & a_{j,j} + (x - x_0)\Big(a_{j+1,j+1} + (x - z)\Big(a_{j+2,j+1} + \sum_{i=j+3}^{N} a_{i,j+1}\omega_{i-j-2}(x)\Big)\Big) = \\ = \; & a_{j,j} + (x - x_0)(a_{j+1,j} + \ldots + (x - x_{N-j-2})(a_{N-1,j+1} + \\ & \qquad + (x - x_{N-j-1} + x_{N-j-1} - z)a_{N,j+1})\ldots) = P_{N,j}(x). \end{aligned}$$

Лемма доказана.

Последовательно дифференцируя равенство (2.22) и полагая $j = 0$ и $x = z$, получаем

$$P_{N,0}^{(k)}(z) = k!P_{N,k}(z), \quad k = 0, 1, \ldots, N.$$

Так как $L_N \equiv P_{N,0}$, то равенство (2.16) можно переписать в виде:

$$L_N(x) = P_{N,0}(z) + (x - z)P_{N,1}(z) + \ldots + (x - z)^N P_{N,N}(z),$$

где $P_{N,j}(z) = a_{j,j+1}$, $j = 0, 1, \ldots, N$ и $a_{N,N+1} = a_{N,0}$.

Таким образом, справедливо следующее утверждение:

Теорема 2.2. *Пусть L_N – многочлен вида* (2.11) *и требуется вычислить значение его производной $L_N^{(l)}(z)$, $0 \le l \le N$.*

Переобозначим $a_{k,0} = c_k$, $k = 0, \ldots, N$ и образуем многочлены

$$P_{N,j}(x) = a_{jj} + \sum_{i=j+1}^{N} a_{ij}\omega_{i-j-1}(x), \quad j = 0, 1, \ldots, l,$$

коэффициенты которых находятся по формулам:

$$\begin{aligned} a_{N,j+1} &= a_{N,0}, \\ a_{i,j+1} &= a_{ij} + (x - x_{i-j})a_{i+1,j+1}, \quad i = N-1, \ldots, j. \end{aligned}$$

Тогда

$$L_N^{(l)}(z)/l! = P_{N,l}(z) = a_{l,l+1}, \quad 0 \le l \le N.$$

Приведенный алгоритм может быть легко запрограммирован. Приведем фрагменты соответствующей программы на языке Матлаб. Предположим, что имеется два массива данных x и f каждый длины $n+1$. Вначале, согласно формуле (2.12), вычисляем разделенные разности:

```
for i = 1 : n + 1  a(i) = f(i); end
for i = 1 : n
   for j = n + 1 : -1 : i + 1
      a(j) = (a(j) - a(j - 1))/(x(j) - x(j - i));
   end
end
```

Вычисление разделенных разностей может быть выполнено и другим путем:

```
a(n + 1) = f(n + 1);
for i = n : -1 : 1
   a(i) = f(i);
      for j = i + 1 : n + 1
         a(j) = (a(j) - a(j - 1))/(x(j) - x(i));
      end
end
```

Теперь для нахождения значения $L_N^{(l)}(z)$, $0 \le l \le N$ можно воспользоваться следующими циклами:

```
for i = 1 : n + 1  d(i) = a(i); end
k = 1;
for i = 1 : l + 1
   if (i > 1)  k = k * (i − 1); end
      if (i < n + 1)
         for j = n : −1 : i
            d(j) = d(j)+(z−x(j−i+1))∗d(j+1);
         end
      end
end
vnewn = k * d(l + 1);
```

Отметим, что если $x_i = 0$ для всех i, то многочлен L_N в равенстве (2.11) принимает вид $L_N(x) = c_0 + c_1 x + \ldots + c_N x^N$ и описанный алгоритм сводится к общеизвестному алгоритму синтетического деления.

Пример 2.2. Функция $f(x) = 1/(1+x^2)$ задана своими значениями в табл. 2.1. По данным табл. 2.1 построить интерполяционный кубический многочлен Ньютона L_3. По схеме Горнера вычислить значения $L_3(1{,}5)$ и $L_3'(1{,}5)$. Оценить погрешность полученных аппроксимаций.

Таблица 2.1

i	x_i	f_i
0	-1	0,5
1	0	1,0
2	1	0,5
3	2	0,2

Решение. По данным табл. 2.1, пользуясь формулами (2.12), построим вначале табл. 2.2 разделенных разностей.

Таблица 2.2

x_i	f_i	$f[x_i, x_{i+1}]$	$f[x_i, x_{i+1}, x_{i+2}]$	$f[x_i, \ldots, x_{i+3}]$
-1	0,5			
0	1,0	0,5		
1	0,5	-0,5	-0,5	
2	0,2	-0,3	0,1	0,2

Согласно равенству (2.11), интерполяционный кубический многочлен Ньютона L_3 принимает вид:

$$L_3(x) \;=\; c_0 + c_1(x - x_0) + c_2(x - x_0)(x - x_1) +$$

$$
\begin{aligned}
&+c_3(x-x_0)(x-x_1)(x-x_3) = \\
&= \quad 0{,}5 + 0{,}5(x+1) - 0{,}5(x+1)x + 0{,}2(x+1)x(x-1).
\end{aligned}
$$

Вводя переобозначения $a_{i,0} = c_i$, $i = 0, \ldots, 3$, перепишем многочлен L_3 в виде:

$$
\begin{aligned}
L_3(x) \equiv P_{3,0}(x) &= a_{0,0} + (x-x_0)(a_{1,0} + (x-x_1)(a_{2,0} + (x-x_2)a_{3,0})) = \\
&= 0{,}5 + (x+1)(0{,}5 + x(-0{,}5 + (x-1)0{,}2)).
\end{aligned}
$$

Значение многочлена $P_{3,0}$ в точке $z = 1{,}5$ находим по схеме Горнера:

$$
\begin{aligned}
a_{3,1} &= a_{3,0} = 0{,}2; \\
a_{2,1} &= a_{2,0} + (z-x_2)a_{3,1} = -0{,}5 + 0{,}5 \cdot 0{,}2 = -0{,}4; \\
a_{1,1} &= a_{1,0} + (z-x_1)a_{2,1} = 0{,}5 + 1{,}5(-0{,}4) = -0{,}1; \\
a_{0,1} &= a_{0,0} + (z-x_0)a_{1,1} = 0{,}5 + 2{,}5(-0{,}1) = 0{,}25.
\end{aligned}
$$

Таким образом, $L_3(1{,}5) \equiv P_{3,0}(1{,}5) = 0{,}25$.

Для нахождения значения производной $L_3'(1{,}5)$ выпишем многочлен

$$
\begin{aligned}
P_{3,1}(x) &= a_{1,1} + (x-x_0)(a_{2,1} + (x-x_1)a_{3,1}) = \\
&= -0{,}1 + (x+1)(-0{,}4 + x \cdot 0{,}2).
\end{aligned}
$$

Вычисления опять проводим по схеме Горнера:

$$
\begin{aligned}
a_{3,2} &= a_{3,1} = 0{,}2; \\
a_{2,2} &= a_{2,1} + (z-x_1)a_{3,2} = -0{,}4 + 1{,}5 \cdot 0{,}2 = -0{,}1; \\
a_{1,2} &= a_{1,1} + (z-x_0)a_{2,2} = -0{,}1 + 2{,}5(-0{,}1) = -0{,}35.
\end{aligned}
$$

Следовательно, $L_3'(1{,}5) = P_{3,1}(1{,}5) = -0{,}35$.

Для функции $f(x) = 1/(1+x^2)$ находим погрешность полученных аппроксимаций:

$$
\begin{aligned}
f(1{,}5) - L_3(1{,}5) &= 0{,}30769 - 0{,}25 = 0{,}05769, \\
f'(1{,}5) - L_3'(1{,}5) &= -0{,}28402 + 0{,}35 = 0{,}06598.
\end{aligned}
$$

§ 2.5. Сходимость интерполяционного процесса

Выбор многочленов в качестве аппарата аппроксимации функций обычно мотивируется следующей известной теоремой Вейерштрасса.

Теорема 2.3. *Если f – непрерывная функция на отрезке $[a, b]$, то для всякого $\varepsilon > 0$ существует многочлен P_N степени $N = N(\varepsilon)$ такой, что*

$$
\max_{a \le x \le b} |f(x) - P_N(x)| < \varepsilon.
$$

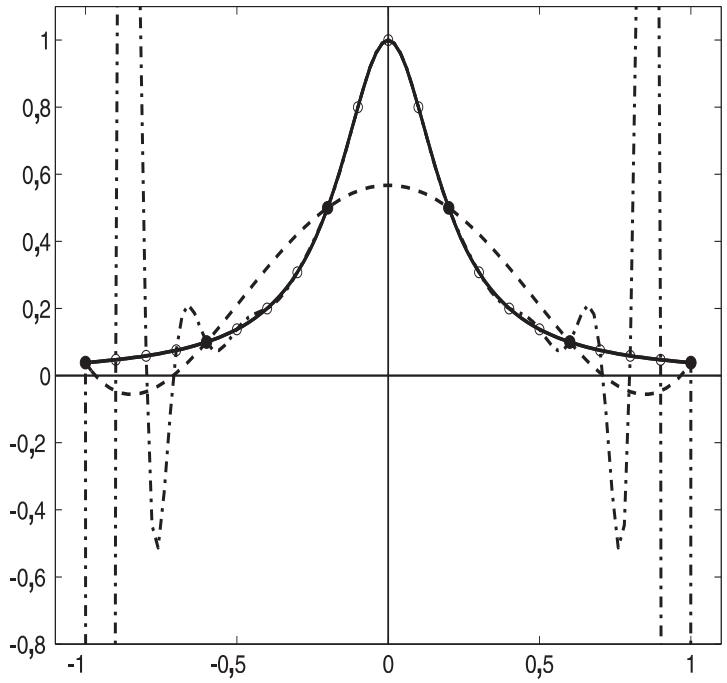


Рис. 2.2. Интерполяция функции Рунге (сплошная линия) многочленами 5-й (штриховая линия) и 20-й (штрих-пунктирная линия) степеней по равноотстоящим узлам. Черными точками и кружками указаны соответствующие этим многочленам точки интерполяции.

Нас интересует, однако, задача интерполяции и, в частности, сходимость интерполяционного процесса, т. е. если f – непрерывная функция на $[a,b]$ и выполняются условия интерполяции $P_N(x_i) = f(x_i)$, $i = 0, 1, \ldots, N$, то будет ли при $N \to \infty$ стремиться к нулю величина $\max_{a \le x \le b} |f(x) - P_N(x)|$?

Можно привести примеры, когда сходимости нет. Наиболее известным является пример Рунге. Пусть функция Рунге $f(x) = 1/(1 + 25x^2)$ интерполируется на отрезке $[-1, 1]$ многочленом P_N по равноотстоящим узлам $x_i = -1 + 2i/N$, $i = 0, 1, \ldots, N$. Тогда оказывается, что

$$\lim_{N \to \infty} \max_{0{,}726\ldots \le |x| < 1} |f(x) - P_N(x)| = \infty.$$

Рис. 2.2 иллюстрирует расходимость интерполяционного процесса для примера Рунге. На рис. 2.2 интерполяционный многочлен P_{20} вблизи концов отрезка $[-1, 1]$ существенно отклоняется от интерполируемой функции. Осцилляции стремятся к бесконечности при возрастании N.

Более того, имеет место следующий результат Фабера (см. [?]).

Теорема 2.4. *Для всякой заданной системы узлов интерполяции*

$$a \le x_0^{(N)} < x_1^{(N)} < \ldots < x_N^{(N)} \le b \quad (N \ge 0) \tag{2.23}$$

существует непрерывная на $[a,b]$ *функция* f *такая, что последовательность*

интерполяционных многочленов по этой системе узлов не сходится равномерно к f при $N \to \infty$.

Сходимость интерполяционного процесса можно, однако, обеспечить за счет специального выбора узлов интерполяции.

Теорема 2.5 (см. [?]). *Если f – непрерывная на $[a, b]$ функция, то существует такая система узлов интерполяции вида* (2.23)*, что последовательность интерполяционных многочленов P_N по этой системе равномерно сходится к f, т. е.*

$$\lim_{N\to\infty} \max_{a\le x\le b} |P_N(x) - f(x)| = 0.$$

Множество непрерывных функций является слишком широким, чтобы для него существовала единственная таблица узлов, обеспечивающая равномерную сходимость интерполяционного процесса для всех функций множества. В качестве гарантирующей сходимость к функции с ограниченной производной системы узлов интерполяции обычно выбираются нули многочленов Чебышева:

$$x_j = \frac{a+b}{2} + \frac{b-a}{2} \cos \frac{\pi[2(N-j)+1]}{2(N+1)}, \quad j = 0, 1, \ldots, N.$$

Проблема сходимости интерполяционного процесса, однако, как правило, исчезает при переходе к интерполяции кусочными многочленами Лагранжа, называемыми *лагранжевыми сплайнами.*

§ 2.6. Кусочно-линейная интерполяция

Простейшим примером лагранжева сплайна, всегда гарантирующего сходимость интерполяционного процесса к интерполируемой функции, является кусочно-линейная интерполяция. В этом случае на отрезке $[x_i, x_{i+1}]$, $i = 0, 1, \ldots, N-1$ имеем интерполяционный многочлен Лагранжа первой степени

$$L_{i,1}(x) = f_i \frac{x_{i+1} - x}{h_i} + f_{i+1} \frac{x - x_i}{h_i}, \quad h_i = x_{i+1} - x_i. \tag{2.24}$$

Таким образом, на всем отрезке $[a, b]$ имеем набор из N интерполяционных многочленов Лагранжа первой степени, образующих *лагранжев сплайн первой степени.*

Множество ломаных с узлами на разбиении $\Delta : a = x_0 < x_1 < \ldots < x_N = b$ обозначим через $S_1(\Delta)$. Очевидно, что элементами этого множества являются упорядоченные наборы многочленов первой степени, состыкованных между собой в узлах сетки Δ таким образом, что они образуют непрерывную функцию. Обычные операции сложения элементов из множества $S_1(\Delta)$ и их умножения на вещественные числа дают элементы того же множества, которое, таким образом, является линейным пространством.

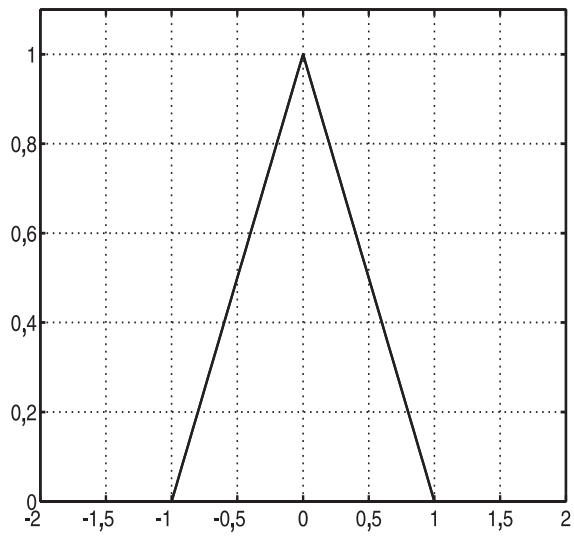


Рис. 2.3. Базисный сплайн первой степени с узлами на целочисленной сетке.

Каждый из N составляющих ломаную S многочленов имеет два коэффициента, что в совокупности дает $2N$ параметров. Вычитая из этого числа $N-1$ ограничений стыковки соседних многочленов, заключаем, что размерность линейного пространства $S_1(\Delta)$ равна $N+1$. Таким образом, условия интерполяции $S(x_i) = f_i$, $i = 0, 1, \ldots, N$ однозначно определяют ломаную S_f, которая совпадает с лагранжевым сплайном первой степени: $S_f \equiv L_{i,1}$ на $[x_i, x_{i+1}]$, $i = 0, 1, \ldots, N-1$.

В пространстве $S_1(\Delta)$ построим базис из функций с конечными носителями минимальной длины. С этой целью расширим сетку Δ, добавив точки $x_{-1} < a$ и $x_{N+1} > b$. Базисные сплайны (сокращенно B-сплайны) первой степени определим по формуле (рис. 2.3 для случая целочисленной сетки)

$$B_{j,1}(x) = \begin{cases} \dfrac{x - x_j}{h_j}, & x_j \le x < x_{j+1}; \\ \dfrac{x_{j+2} - x}{h_{j+1}}, & x_{j+1} \le x < x_{j+2}; \\ 0 \qquad \text{в противном случае.} \end{cases} \tag{2.25}$$

Отсюда непосредственно следует, что функции-крыши $B_{j,1}$ обладают следующими свойствами:

$$B_{j,1}(x) \begin{cases} > 0, & \text{если } x_j < x < x_{j+2}; \\ \equiv 0 & \text{в противном случае.} \end{cases}$$

$$\sum_{j=-1}^{N-1} B_{j,1}(x) \equiv 1 \quad \text{для} \quad x \in [a, b].$$

Теорема 2.6. *Функции $B_{j,1}$, $j = -1, 0, \ldots, N-1$ линейно независимы и образуют базис в пространстве $S_1(\Delta)$.*

Доказательство. Предположим противное, т. е. найдутся не все равные нулю постоянные c_i такие, что

$$c_{-1}B_{-1,1}(x) + \ldots + c_{N-1}B_{N-1,1}(x) \equiv 0 \quad \text{для} \quad x \in [a,b].$$

Функции $B_{j,1}$ имеют конечные носители. Поэтому с учетом формулы (2.25) на отрезке $[x_i, x_{i+1}]$ имеем

$$\sum_{j=-1}^{N-1} c_j B_{j,1}(x) = c_{i-1}B_{i-1,1}(x) + c_i B_{i,1}(x) = c_{i-1}\frac{x_{i+1}-x}{h_i} + c_i\frac{x-x_i}{h_i} = 0.$$

Полагая здесь $x = x_i$ и $x = x_{i+1}$, получаем $c_{i-1} = c_i = 0$. Продолжая этот процесс, находим, что все c_i равны нулю.

Так как функции $B_{j,1}$ являются элементами пространства $S_1(\Delta)$ и число их равно размерности этого пространства, то они образуют в нем базис. Теорема доказана.

Пример 2.3. Функция f приближается на отрезке $[a,b]$ с помощью кусочно-линейной интерполяции по узлам $x_i = a + i(b-a)/N$, $i = 0, 1, \ldots, N$, причем ошибки округления при вычислении значений f не превышают заданного $\varepsilon > 0$. Сколько нужно взять узлов интерполяции, чтобы обеспечить точность приближения E ($\varepsilon < E$)?

Решение. Пусть $f(x_i) = \bar{f}_i + \varepsilon_i$, $i = 0, 1, \ldots, N$, где ε_i – ошибка округления. Положим $h = (b-a)/N$. Согласно формуле (2.24), на подотрезке $[x_i, x_{i+1}]$ имеем

$$\begin{aligned} |f(x) - L_{i,1}(x)| &= \Big| f(x) - \bar{f}_i\frac{x_{i+1}-x}{h} + \bar{f}_{i+1}\frac{x-x_i}{h}\Big| = \\ &= \Big| f(x) - f(x_i)\frac{x_{i+1}-x}{h} - f(x_{i+1})\frac{x-x_i}{h} + \\ &\quad + \varepsilon_i\frac{x_{i+1}-x}{h} + \varepsilon_{i+1}\frac{x-x_i}{h}\Big|. \end{aligned}$$

Пользуясь теперь формулой (2.5) для $N = 1$, получаем неравенство

$$\begin{aligned} |f(x) - L_{i,1}(x)| &\le \frac{1}{2}(x_{i+1}-x)(x-x_i)|f''(\xi_x)| + \max(|\varepsilon_i|, |\varepsilon_{i+1}|) \le \\ &\le \frac{h^2}{8}M + \varepsilon \le E, \quad M = \max_{a\le x\le b}|f''(x)|. \end{aligned}$$

Отсюда следует оценка

$$\frac{1}{8}\Big(\frac{b-a}{N}\Big)^2 M \le E - \varepsilon \quad \text{или} \quad N \ge \frac{b-a}{4}\Big(\frac{2M}{E-\varepsilon}\Big)^{1/2}.$$

Пример 2.4. Пусть выполнены условия примера 2.3. При каком числе узлов интерполяции ошибка приближения f' на $[a,b]$ будет минимальной?

Решение. Согласно формуле (2.24) на отрезке $[x_i, x_{i+1}]$ имеем

$$f'(x) - L'_{i,1}(x) = f'(x) - \frac{\bar{f}_{i+1} - \bar{f}_i}{h} =$$

$$= f'(x) - \frac{f(x_{i+1}) - f(x_i)}{h} + \frac{\varepsilon_{i+1} - \varepsilon_i}{h}.$$

Пользуясь разложением по формуле Тейлора, получаем:

$$\begin{aligned} f(x_i) &= f(x) + f'(x)(x_i - x) + f''(\xi_1)\frac{(x_i - x)^2}{2}, \qquad \xi_1 \in (x_i, x), \\ f(x_{i+1}) &= f(x) + f'(x)(x_{i+1} - x) + f''(\xi_2)\frac{(x_{i+1} - x)^2}{2}, \quad \xi_2 \in (x, x_{i+1}). \end{aligned}$$

Отсюда

$$\frac{f(x_{i+1}) - f(x_i)}{h} = f'(x) + f''(\xi_2)\frac{(x_{i+1} - x)^2}{2h} - f''(\xi_1)\frac{(x_i - x)^2}{2h},$$

что дает оценку

$$\left| f'(x) - \frac{f(x_{i+1}) - f(x_i)}{h} \right| \le \frac{(x - x_i)^2 + (x_{i+1} - x)^2}{2h} \max_{x_i \le x \le x_{i+1}} |f''(x)|.$$

Тогда

$$\begin{aligned} |f'(x) - L'_{i,1}(x)| &\le \frac{h}{2} \max_{x_i \le x \le x_{i+1}} |f''(x)| + \frac{|\varepsilon_i| + |\varepsilon_{i+1}|}{h} \le \\ &\le \frac{h}{2} M + \frac{2\varepsilon}{h} = \varphi(h, \varepsilon). \end{aligned}$$

Функция φ достигает минимума по переменной h в точке $\varphi'(h, \varepsilon) = 0$, откуда $h = 2(\varepsilon/M)^{1/2}$. Поэтому оптимальное число узлов сетки должно быть равно минимальному целому числу $N \ge (M/\varepsilon)^{1/2}(b - a)/2$.

§ 2.7. Интерполяция кубическими лагранжевыми сплайнами

Точность аппроксимации можно существенно повысить, перейдя к интерполяции кусочными многочленами Лагранжа более высоких степеней: кусочно-квадратическими, кусочно-кубическими и т. д. На практике наиболее употребительны кусочно-кубические многочлены Лагранжа. В этом случае в предположении наличия данных (x_i, f_i), $i = -1, 0, \ldots, N + 1$ на каждом из подотрезков $[x_i, x_{i+1}]$, $i = 0, 1, \ldots, N - 1$ берется отдельный кубический многочлен Лагранжа:

$$L_{i,3}(x) = \sum_{j=i-1}^{i+2} f_j \frac{\omega_{i-1,3}(x)}{(x - x_j)\omega'_{i-1,3}(x_j)}, \quad \omega_{i-1,3}(x) = \prod_{j=i-1}^{i+2} (x - x_j). \tag{2.26}$$

В целом на отрезке $[a, b]$ имеем набор из N кубических многочленов Лагранжа, образующих непрерывную функцию, называемую *кубическим лагранжевым сплайном*. Если данные (x_j, f_j), $j = -1, N+1$ отсутствуют, то можно рассмотреть многочлен $L_{1,3}$ на отрезке $[x_0, x_2]$ и $L_{N-2,3}$ – на $[x_{N-2}, x_N]$. При этом, однако, точность аппроксимации на отрезках $[x_0, x_1]$ и $[x_{N-1}, x_N]$ несколько снижается (см. [13]).

Пользуясь формулой (2.5) для $N = 3$, на отрезке $[x_i, x_{i+1}]$ имеем

$$|f(x) - L_{i,3}(x)| \le \frac{1}{4!}|\omega_{i-1,3}(x)| \max_{x_{i-1} \le x \le x_{i+2}} |f^{(4)}(x)| \le \frac{9}{384}\bar{h}^4 \|f^{(4)}\|_{C[a,b]}, \qquad (2.27)$$

где $\bar{h} = \max_i h_i$ и $\|f\|_{C[a,b]} = \max_{a \le x \le b} |f(x)|$.

К сожалению, на редкой сетке график кубического лагранжева сплайна может иметь изломы, так как производные соседних многочленов не состыкованы между собой. Исключение составляет случай равномерной сетки ($h_i = h$ для всех i), когда вторая производная кубического лагранжева сплайна оказывается непрерывной.

Запишем два соседних кубических многочлена Лагранжа:

$$\begin{aligned} L_{i-1,3}(x) &= f_{i-1} + f[x_{i-1}, x_i](x - x_{i-1}) + \\ &\quad + f[x_{i-1}, x_i, x_{i+1}](x - x_{i-1})(x - x_i) + \\ &\quad + f[x_{i-2}, x_{i-1}, x_i, x_{i+1}](x - x_{i-1})(x - x_i)(x - x_{i+1}), \\ L_{i,3}(x) &= f_{i-1} + f[x_{i-1}, x_i](x - x_{i-1}) + \\ &\quad + f[x_{i-1}, x_i, x_{i+1}](x - x_{i-1})(x - x_i) + \\ &\quad + f[x_{i-1}, x_i, x_{i+1}, x_{i+2}](x - x_{i-1})(x - x_i)(x - x_{i+1}). \end{aligned}$$

Беря разность этих многочленов, имеем

$$L_{i,3}(x) - L_{i-1,3}(x) = \theta_{i,4}(x - x_{i-1})(x - x_i)(x - x_{i+1}), \qquad (2.28)$$

где $\theta_{i,4} = (x_{i+2} - x_{i-2}) f[x_{i-2}, \ldots, x_{i+2}]$.

Дифференцируя полученное равенство и полагая $x = x_i$, находим

$$\begin{aligned} L'_{i,3}(x_i) - L'_{i-1,3}(x_i) &= -h_{i-1}h_i\theta_{i,4}, \\ L''_{i,3}(x_i) - L''_{i-1,3}(x_i) &= (h_{i-1} - h_i)\theta_{i,4}. \end{aligned}$$

Таким образом, при $h_{i-1} = h_i$ имеем $L''_{i-1,3}(x_i) = L''_{i,3}(x_i)$. Отсюда следует непрерывность второй производной кубического лагранжева сплайна на равномерной сетке Δ.

Применим простые приемы, чтобы показать как можно гладко состыковать между собой соседние кубические многочлены Лагранжа для получения кривой класса C^2, обеспечивающей порядок аппроксимации $O(\overline{h}^4)$.

§ 2.8. Локальная аппроксимация кубическими сплайнами

На отрезке $[x_i, x_{i+1}]$, $i = 0, 1, \ldots, N-1$ рассмотрим «подправленный» кубический многочлен Лагранжа:

$$S_{i,3}(x) = L_{i,3}(x) + C_{i,1}(x - x_i)^3 + C_{i,2}(x_{i+1} - x)^3.$$

Потребуем, чтобы:

$$S^{(r)}_{i-1,3}(x_i - 0) = S^{(r)}_{i,3}(x_i + 0), \quad r = 0, 1, 2, \quad i = 1, 2, \ldots, N-1. \tag{2.29}$$

Беря разность соседних многочленов $S_{i,3}$ и $S_{i-1,3}$ и учитывая равенство (2.28), имеем

$$\begin{aligned} S_{i,3}(x) - S_{i-1,3}(x) &= \theta_{i,4}(x - x_{i-1})(x - x_i)(x - x_{i+1}) + \\ &+ (C_{i,1} + C_{i-1,2})(x - x_i)^3 + C_{i,2}(x_{i+1} - x)^3 - C_{i-1,1}(x - x_{i-1})^3. \end{aligned}$$

Отсюда, согласно условиям (2.29), получаем систему уравнений

$$\begin{cases} h^3_{i-1}C_{i-1,1} - h^3_i C_{i,2} = 0; \\ 3h^2_{i-1}C_{i-1,1} + 3h^2_i C_{i,2} = -h_{i-1}h_i\theta_{i,4}; \\ 3h_{i-1}C_{i-1,1} - 3h_i C_{i,2} = (h_{i-1} - h_i)\theta_{i,4}. \end{cases} \tag{2.30}$$

Уравнения переопределенной системы (2.30) являются линейно зависимыми. Система имеет единственное решение:

$$C_{i-1,1} = -\frac{h^2_i\theta_{i,4}}{3h_{i-1}(h_{i-1} + h_i)}, \quad C_{i,2} = \Big(\frac{h_{i-1}}{h_i}\Big)^3 C_{i-1,1}.$$

Таким образом, на отрезке $[x_i, x_{i+1}]$ гладкий кубический лагранжев сплайн принимает вид:

$$S_{i,3}(x) = L_{i,3}(x) - \frac{h^2_{i+1}\theta_{i+1,4}}{3h_i(h_i + h_{i+1})}(x - x_i)^3 - \frac{h^2_{i-1}\theta_{i,4}}{3h_i(h_{i-1} + h_i)}(x_{i+1} - x)^3. \tag{2.31}$$

Формула (2.31) использует шесть точек исходных данных (x_j, f_j), $j = i - 2, \ldots, i + 3$ и поэтому несколько сложнее в применении, чем обычный кубический многочлен Лагранжа. Свойство интерполяции потеряно. Вместо него имеет место свойство локальной аппроксимации. Покажем, однако, что точность приближения фактически сохраняется.

Согласно равенству (2.14), формулу (2.31) можно переписать в виде:

$$\begin{aligned} f(x) - S_{i,3}(x) &= f[x_{i-1}, \ldots, x_{i+2}, x]\omega_{i-1,4}(x) + \frac{h^2_{i+1}\theta_{i+1,4}}{3h_i(h_i + h_{i+1})}(x - x_i)^3 + \\ &+ \frac{h^2_{i-1}\theta_{i,4}}{3h_i(h_{i-1} + h_i)}(x_{i+1} - x)^3 = \\ &= \Big[\omega_{i-1,4}(x) + \frac{h^2_{i+1}(x_{i+3} - x_{i-1})}{3h_i(h_i + h_{i+1})}(x - x_i)^3 + \frac{h^2_{i-1}(x_{i+2} - x_{i-2})}{3h_i(h_{i-1} + h_i)} \times \\ &\times (x_{i+1} - x)^3\Big] f[x_{i-1}, \ldots, x_{i+2}, \xi], \quad \xi \in [x_{i-2}, x_{i+3}]. \end{aligned}$$

Отсюда для $x \in [x_i, x_{i+1}]$ имеем оценку:

$$|f(x) - S_{i,3}(x)| \leq \left[t^2(1-t)^2 + \frac{2}{3}\right]\bar{h}_i^4 \max_{x_{i-2}\leq\xi\leq x_{i+3}} |f[x_{i-1}, \ldots, x_{i+2}, \xi]| \leq$$
$$\leq \frac{35}{48}\bar{h}_i^4 \max_{x_{i-2}\leq\xi\leq x_{i+3}} |f[x_{i-1}, \ldots, x_{i+2}, \xi]|, \quad \bar{h}_i = \max_{|i-j|\leq 2} h_j.$$

Используя равенство (2.15), эту оценку можно также переписать в виде

$$|f(x) - S_{i,3}(x)| \leq \left[t^2(1-t)^2 + \frac{2}{3}\right]\frac{\bar{h}_i^4}{24}M \leq \frac{35}{1152}\bar{h}^4 M, \tag{2.32}$$

где $M = \|f^{(4)}\|_{C[a,b]}$.

Сравнивая теперь оценки (2.27) и (2.32), делаем вывод, что переход от интерполяции кубическим лагранжевым сплайном к локальной аппроксимации не приводит к потере точности приближения (по сравнению с оценкой (2.27) константа в оценке (2.32) увеличивается незначительно).

Изложение общих методов построения локальных аппроксимаций для сплайнов произвольной степени дается в работе [13].

§ 2.9. Интерполяционный кубический сплайн

Весьма эффективным методом решения задачи интерполяции является построение интерполяционного кубического сплайна. Достаточные для многих приложений свойства гладкости таких сплайнов сочетаются с простотой их компьютерной реализации и высокой точностью получаемых результатов.

Определение 2.1. *Функция S называется кубическим сплайном, если:*

а) на каждом отрезке $[x_i, x_{i+1}]$ функция S является кубическим многочленом, т. е.

$$S(x) \equiv S_i(x) = a_{i,0} + a_{i,1}(x - x_i) + a_{i,2}(x - x_i)^2 + a_{i,3}(x - x_i)^3$$
$$\text{для} \quad x \in [x_i, x_{i+1}], \quad i = 0, 1, \ldots, N-1;$$

б) соседние многочлены гладко состыкованы между собой:

$$S_{i-1}^{(r)}(x_i - 0) = S_i^{(r)}(x_i + 0), \quad i = 1, 2, \ldots, N-1, \quad r = 0, 1, 2.$$

Кубический сплайн S называется *интерполяционным*, если он удовлетворяет условиям:

$$S(x_i) = f_i, \quad i = 0, 1, \ldots, N.$$

Точки стыковки x_i, $i = 1, 2, \ldots, N-1$, составляющих сплайн S соседних многочленов, называются *узлами* сплайна. Они могут не совпадать с точками интерполяции. Узлы сплайна могут также иметь различную кратность в зависимости от числа сопрягаемых производных. В частности, узел x_i имеет кратность k_i

$(0 \le k_i \le 3)$, если в этой точке разрывны k_i старших производных сплайна (производные соседних многочленов сопрягаются в этом узле до порядка $3-k_i$). Здесь мы будем рассматривать только наиболее употребительные на практике сплайны с простыми узлами (кратности 1).

Множество кубических сплайнов, удовлетворяющих определению 2.1, обозначим через $S_3(\Delta)$. Очевидно, что оно состоит из упорядоченных наборов кубических многочленов, гладко состыкованных между собой так, что они образуют дважды непрерывно дифференцируемую функцию. Обычные операции сложения элементов из $S_3(\Delta)$ и их умножения на вещественные числа дают опять элементы множества $S_3(\Delta)$, которое, таким образом, является линейным пространством.

Каждый из N составляющих кубический сплайн S многочленов имеет 4 коэффициента, что в совокупности дает $4N$ параметров. Из этого числа следует вычесть $3(N-1)$ ограничений гладкости. Поэтому линейное пространство $S_3(\Delta)$ имеет размерность $N+3$. Вычитая теперь $N+1$ условий интерполяции, получаем два свободных параметра, которые обычно определяются с помощью ограничений на значения сплайна и его производных на концах отрезка $[a,b]$ (или вблизи концов). Эти ограничения называются *краевыми условиями*. Существует несколько различных видов краевых условий, из которых наиболее распространенными являются такие типы:

1. ограничения на значения первой производной сплайна:

$$S'(x_0) = f'_0 \quad \text{и} \quad S'(x_N) = f'_N;$$

2. ограничения на значения второй производной сплайна:

$$S''(x_0) = f''_0 \quad \text{и} \quad S''(x_N) = f''_N;$$

3. периодические краевые условия:

$$S^{(r)}(x_0) = S^{(r)}(x_N), \quad r = 0, 1, 2;$$

4. тождественное совпадение ближайших к концам отрезка $[a,b]$ соседних многочленов: $S_0(x) \equiv S_1(x)$ и $S_{N-2}(x) \equiv S_{N-1}(x)$, т. е. $S'''(x_i - 0) = S'''(x_i + 0)$, $i = 1, N-1$.

Выполнения периодических краевых условий естественно требовать в предположении, что интерполируемая функция f – периодическая с периодом $b-a$.

§ 2.10. Алгоритм построения интерполяционного кубического сплайна

Вторая производная кубического сплайна S'' является непрерывной кусочно-линейной функцией. Поэтому, полагая $M_i = S''(x_i)$, $i = 0, 1, \ldots, N$ и $h_i = x_{i+1} - x_i$,

$i = 0, 1, \ldots, N-1$, можно записать

$$S''(x) \equiv S_i''(x) = M_i \frac{x_{i+1} - x}{h_i} + M_{i+1} \frac{x - x_i}{h_i}, \quad x \in [x_i, x_{i+1}]. \tag{2.33}$$

Повторное интегрирование формулы (2.33) дает выражение для кубического многочлена S_i, содержащее две произвольных постоянных:

$$S_i(x) = M_i \frac{(x_{i+1} - x)^3}{6h_i} + M_{i+1} \frac{(x - x_i)^3}{6h_i} + C_{i,1}(x_{i+1} - x) + C_{i,2}(x - x_i). \tag{2.34}$$

Подставляя сюда последовательно $x = x_i$ и $x = x_{i+1}$ и используя условия интерполяции $S_i(x_i) = f_i$ и $S_i(x_{i+1}) = f_{i+1}$, находим:

$$M_i \frac{h_i^2}{6} + C_{i,1} h_i = f_i, \quad M_{i+1} \frac{h_i^2}{6} + C_{i,2} h_i = f_{i+1}.$$

Выражая из этих уравнений постоянные $C_{i,1}$ и $C_{i,2}$ и подставляя их в формулу (2.34), получаем формулу кубического сплайна на подотрезке $[x_i, x_{i+1}]$:

$$\begin{aligned} S_i(x) \;=\; & M_i \frac{(x_{i+1} - x)^3}{6h_i} + M_{i+1} \frac{(x - x_i)^3}{6h_i} + \Big(f_i - M_i \frac{h_i^2}{6}\Big) \frac{x_{i+1} - x}{h_i} + \\ & + \Big(f_{i+1} - M_{i+1} \frac{h_i^2}{6}\Big) \frac{x - x_i}{h_i}. \end{aligned} \tag{2.35}$$

Для нахождения неизвестных коэффициентов M_i, $i = 0, 1, \ldots, N$ используем непрерывность первой производной сплайна S'. Согласно формуле (2.35), имеем

$$\begin{aligned} S_i'(x) \;=\; & -M_i \frac{(x_{i+1} - x)^2}{2h_i} + M_{i+1} \frac{(x - x_i)^2}{2h_i} - \Big(\frac{f_i}{h_i} - M_i \frac{h_i}{6}\Big) + \\ & + \Big(\frac{f_{i+1}}{h_i} - M_{i+1} \frac{h_i}{6}\Big). \end{aligned} \tag{2.36}$$

Подставляя сюда $x = x_i$, находим

$$S_i'(x_i + 0) = -M_i \frac{h_i}{3} - M_{i+1} \frac{h_i}{6} + f[x_i, x_{i+1}].$$

Выражение для S_{i-1}' получим, заменяя в формуле (2.36) индекс i на $i - 1$. Подставляя в него $x = x_i$, имеем

$$S_{i-1}'(x_i - 0) = M_{i-1} \frac{h_{i-1}}{6} + M_i \frac{h_{i-1}}{3} + f[x_{i-1}, x_i].$$

Теперь из условия $S_{i-1}'(x_i - 0) = S_i'(x_i + 0)$, $i = 1, 2, \ldots, N-1$, получаем при обозначении $\delta_i f = f[x_i, x_{i+1}] - f[x_{i-1}, x_i]$ систему линейных уравнений:

$$h_{i-1} M_{i-1} + 2(h_{i-1} + h_i) M_i + h_i M_{i+1} = 6\delta_i f, \quad i = 1, 2, \ldots, N-1. \tag{2.37}$$

Система (2.37) является недоопределенной, так как содержит $N-1$ уравнений для нахождения $N+1$ неизвестных. Для замыкания этой системы используем приведенные в § 2.9 краевые условия.

Используя формулы (2.33) и (2.36), можно переписать перечисленные в разделе 2.9 краевые условия в виде:

1. $2M_0 + M_1 = \dfrac{6}{h_0}(f[x_0, x_1] - f'_0)$,

   $M_{N-1} + 2M_N = \dfrac{6}{h_{N-1}}(f'_N - f[x_{N-1}, x_N])$;

2. $M_0 = f''_0$ и $M_N = f''_N$;

3. $f_{N+i} = f_i$, $M_{N+i} = M_i$, $h_{N+i} = h_i$ для всех i;

4. $\dfrac{M_{i+1} - M_i}{h_i} = \dfrac{M_i - M_{i-1}}{h_{i-1}}$, $i = 1, N-1$.

Далее будет показано, что решение системы (2.37) с краевыми условиями типов 1–4 существует и единственно. Найдя это решение, затем вычисления проводим по формуле (2.35).

§ 2.11. Системы линейных уравнений

Рассмотрим более подробно результирующие системы линейных уравнений для вычисления неизвестных M_i, $i = 0, 1, \ldots, N$.

1. Для краевых условий первого типа получаем следующую систему:

$$\begin{pmatrix} 2h_0 & h_0 & 0 & \ldots & 0 \\ h_0 & 2(h_0+h_1) & h_1 & \ldots & 0 \\ \vdots & \ddots & \ddots & \ddots & \vdots \\ 0 & \ldots & h_{N-2} & 2(h_{N-2}+h_{N-1}) & h_{N-1} \\ 0 & \ldots & 0 & h_{N-1} & 2h_{N-1} \end{pmatrix} \begin{pmatrix} M_0 \\ M_1 \\ \vdots \\ M_{N-1} \\ M_N \end{pmatrix} = \bar{b}, \qquad (2.38)$$

где $\bar{b} = \big(6(f[x_0, x_1] - f'_0), 6\delta_1 f, \ldots, 6\delta_{N-1} f, 6(f'_N - f[x_{N-1}, x_N])\big)^T$ и верхний индекс T обозначает операцию транспонирования.

2. Для краевых условий второго типа система отличается лишь первым и последним уравнениями:

$$\begin{pmatrix} 1 & 0 & 0 & \ldots & 0 \\ h_0 & 2(h_0+h_1) & h_1 & \ldots & 0 \\ \vdots & \ddots & \ddots & \ddots & \vdots \\ 0 & \ldots & h_{N-2} & 2(h_{N-2}+h_{N-1}) & h_{N-1} \\ 0 & \ldots & 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} M_0 \\ M_1 \\ \vdots \\ M_{N-1} \\ M_N \end{pmatrix} = \bar{b}, \qquad (2.39)$$

где $\bar{b} = (f''_0, 6\delta_1 f, \ldots, 6\delta_{N-1} f, f''_N)^T$.

3. Для периодических краевых условий уравнение (2.37) можно записать также для $i = N$ (или $i = 0$), т. е.

$$h_{N-1}M_{N-1} + 2(h_{N-1} + h_N)M_N + h_N M_{N+1} = 6\delta_N f. \qquad (2.40)$$

Так как $f_{N+i} = f_i$, $M_{N+i} = M_i$, $i = 0, 1$ и $h_N = h_0$, то

$$f[x_N, x_{N+1}] = \frac{f_{N+1} - f_N}{h_N} = \frac{f_1 - f_0}{h_0} = f[x_0, x_1]$$

и уравнение (2.40) принимает вид:

$$h_0 M_1 + h_{N-1} M_{N-1} + 2(h_{N-1} + h_0) M_N = 6(f[x_0, x_1] - f[x_{N-1}, x_N]).$$

В результате приходим к следующей системе линейных уравнений

$$\begin{pmatrix} 2(h_0+h_1) & h_1 & 0 & \dots & h_0 \\ h_1 & 2(h_1+h_2) & h_2 & \dots & 0 \\ \vdots & \ddots & \ddots & \ddots & \vdots \\ h_0 & \dots & 0 & h_{N-1} & 2(h_{N-1}+h_0) \end{pmatrix} \begin{pmatrix} M_1 \\ M_2 \\ \vdots \\ M_N \end{pmatrix} = \bar{b}, \tag{2.41}$$

где $\bar{b} = \big(6\delta_1 f, \dots, 6\delta_{N-1} f, 6(f[x_0, x_1] - f[x_{N-1}, x_N])\big)^T$.

4. Для краевых условий четвертого типа имеем следующую систему:

$$\begin{pmatrix} h_1 & -(h_0+h_1) & h_0 & \dots & 0 \\ h_0 & 2(h_0+h_1) & h_1 & \dots & 0 \\ \vdots & \ddots & \ddots & \ddots & \vdots \\ 0 & \dots & h_{N-2} & 2(h_{N-2}+h_{N-1}) & h_{N-1} \\ 0 & \dots & h_{N-1} & -(h_{N-2}+h_{N-1}) & h_{N-2} \end{pmatrix} \begin{pmatrix} M_0 \\ M_1 \\ \vdots \\ M_{N-1} \\ M_N \end{pmatrix} = \bar{b}, \tag{2.42}$$

где $\bar{b} = (0, 6\delta_1 f, \dots, 6\delta_{N-1} f, 0)^T$.

Системы (2.38), (2.39) и (2.41) имеют трехдиагональные и «почти» трехдиагональные матрицы. Это позволяет применить для их решения описываемые ниже эффективные алгоритмы типа трехточечной прогонки. Чтобы получить систему с трехдиагональной матрицей в случае краевых условий типа 4, следует предварительно исключить из системы (2.42) неизвестные M_0 и M_N.

Если из второго уравнения системы (2.42), умноженного на h_1, вычесть первое уравнение, умноженное на h_0, то результирующее уравнение принимает вид:

$$(h_0 + 2h_1)(h_0 + h_1) M_1 + (h_1^2 - h_0^2) M_2 = 6h_1 \delta_1 f.$$

Аналогично, если из предпоследнего уравнения системы (2.42), умноженного на h_{N-2}, вычесть последнее уравнение, умноженное на h_{N-1}, то получаем уравнение

$$(h_{N-2}^2 - h_{N-1}^2) M_{N-2} + (2h_{N-2} + h_{N-1})(h_{N-2} + h_{N-1}) M_{N-1} = 6h_{N-2} \delta_{N-1} f.$$

В результате приходим к системе линейных уравнений с трехдиагональной матрицей:

$$\begin{pmatrix} h_0+2h_1 & h_1-h_0 & \dots & 0 \\ h_1 & 2(h_1+h_2) & \dots & 0 \\ \vdots & \ddots & \ddots & \vdots \\ 0 & \dots & h_{N-2}-h_{N-1} & 2h_{N-2}+h_{N-1} \end{pmatrix} \begin{pmatrix} M_1 \\ M_2 \\ \vdots \\ M_{N-1} \end{pmatrix} = \bar{b}, \qquad (2.43)$$

где $\bar{b} = (6\lambda_0\delta_1 f, 6\delta_2 f, \dots, 6\mu_{N-1}\delta_{N-1}f)^T$ при обозначениях $\lambda_0 = h_1/(h_0+h_1)$ и $\mu_{N-1} = h_{N-2}/(h_{N-2}+h_{N-1})$.

§ 2.12. Существование и единственность решения

Рассмотрим вопрос существования и единственности решения систем (2.38), (2.39), (2.41) и (2.43). Эти системы имеют единственные решения тогда и только тогда, когда матрицы этих систем являются невырожденными.

Определение 2.2. *Квадратная матрица $A = \{a_{ij}\}_{i,j=1}^n$ называется матрицей с диагональным преобладанием, если выполняются следующие условия*

$$r_i = |a_{ii}| - \sum_{\substack{j=1 \\ j\neq i}}^{n} |a_{ij}| \geq 0, \quad i = 1, 2, \dots .n. \qquad (2.44)$$

Матрица A называется матрицей со строгим диагональным преобладанием, если неравенства (2.44) *являются строгими.*

Теорема 2.7 (Критерий регулярности Адамара). *Матрица со строгим диагональным преобладанием невырождена.*

Доказательство Предположим противное, т. е. матрица A со строгим диагональным преобладанием является вырожденной. В этом случае $\det(A) = 0$ и однородная система уравнений $Ax = 0$ или

$$\sum_{j=1}^{n} a_{ij}x_j = 0, \quad i = 1, 2, \dots, n,$$

имеет нетривиальное решение $x = (x_1, \dots, x_n)^T$.

Тогда можно найти такое k, что $|x_k| \geq |x_i|$, $i = 1, 2, \dots, n$. Из k-го уравнения однородной системы находим:

$$|a_{kk}|\,|x_k| \leq \sum_{\substack{j=1 \\ j\neq k}}^{n} |a_{kj}|\,|x_j| \leq |x_k| \sum_{\substack{j=1 \\ j\neq k}}^{n} |a_{kj}|.$$

Отсюда

$$|a_{kk}| \leq \sum_{\substack{j=1 \\ j\neq k}}^{n} |a_{kj}|,$$

что противоречит предположению о строгом диагональном преобладании матрицы A. Теорема доказана.

Нетрудно проверить, что системы (2.38), (2.39), (2.41) и (2.43) имеют матрицы со строгим диагональным преобладанием. В случае краевых условий типа 1 из системы (2.38) имеем:

$$\begin{cases} r_0 = 2h_0 - h_0 = h_0 > 0; \\ r_i = 2(h_{i-1} + h_i) - h_{i-1} - h_i = h_{i-1} + h_i > 0, \quad i = 1, 2, \ldots, N-1; \\ r_N = 2h_{N-1} - h_{N-1} = h_{N-1} > 0. \end{cases}$$

Следовательно, матрица этой системы имеет строгое диагональное преобладание.

Для краевых условий типов 2 и 3 из вида уравнений (2.39) и (2.41) также делаем вывод о наличии строгого диагонального преобладания. В случае краевых условий типа 4 вывод о наличии строгого диагонального преобладания получим, анализируя матрицу системы (2.43):

$$\begin{cases} r_1 = h_0 + 2h_1 - |h_1 - h_0| > 0; \\ r_i = 2(h_{i-1} + h_i) - h_{i-1} - h_i = h_{i-1} + h_i > 0, \quad i = 2, 3, \ldots, N-2; \\ r_{N-1} = 2h_{N-2} + h_{N-1} - |h_{N-2} - h_{N-1}| > 0. \end{cases}$$

На основании теоремы 2.7 теперь можно заключить, что системы (2.38), (2.39), (2.41) и (2.43) имеют единственные решения. Поэтому в случае краевых условий типов 1–4 существует единственный интерполяционный кубический сплайн S, удовлетворяющий любому из этих типов краевых условий.

§ 2.13. Метод трехточечной прогонки

Рассмотрим эффективный метод решения систем линейных уравнений, имеющих трехдиагональные матрицы с диагональным преобладанием. Приводимый ниже алгоритм носит название метода трехточечной прогонки и является специальным случаем метода исключения Гаусса. Имея в виду системы линейных уравнений, возникающие при построении интерполяционного кубического сплайна с краевыми условиями типов 1, 2 или 4, рассмотрим следующую линейную систему:

$$\begin{pmatrix} b_1 & c_1 & 0 & \ldots & 0 \\ a_2 & b_2 & c_2 & \ldots & 0 \\ \vdots & \ddots & \ddots & \ddots & \vdots \\ 0 & \ldots & a_{n-1} & b_{n-1} & c_{n-1} \\ 0 & \ldots & 0 & a_n & b_n \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \\ \vdots \\ x_{n-1} \\ x_n \end{pmatrix} = \begin{pmatrix} d_1 \\ d_2 \\ \vdots \\ d_{n-1} \\ d_n \end{pmatrix}. \tag{2.45}$$

Чтобы начать исключение, разделим первое уравнение этой системы на диагональный элемент b_1 и используем обозначения $p_1 = c_1/b_1$ и $q_1 = d_1/b_1$. Предположим, что мы исключили все ненулевые поддиагональные элементы в первых $i-1$ строках. В этом случае система (2.45) преобразуется к виду:

$$\begin{pmatrix} 1 & p_1 & 0 & 0 & 0 & 0 & \dots & 0 \\ 0 & 1 & p_2 & 0 & 0 & 0 & \dots & 0 \\ \vdots & \ddots & \ddots & \ddots & & & & \vdots \\ 0 & \dots & 0 & 1 & p_{i-1} & 0 & \dots & 0 \\ 0 & \dots & 0 & a_i & b_i & c_i & \dots & 0 \\ \vdots & & & & \ddots & \ddots & \ddots & \vdots \\ 0 & \dots & 0 & 0 & 0 & a_{n-1} & b_{n-1} & c_{n-1} \\ 0 & \dots & 0 & 0 & 0 & 0 & a_n & b_n \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \\ \vdots \\ x_{i-1} \\ x_i \\ \vdots \\ x_{n-1} \\ x_n \end{pmatrix} = \begin{pmatrix} q_1 \\ q_2 \\ \vdots \\ q_{i-1} \\ d_i \\ \vdots \\ d_{n-1} \\ d_n \end{pmatrix}.$$

Теперь, чтобы исключить поддиагональный элемент a_i в i-й строке, умножим $(i-1)$-ю строку на a_i и вычтем ее из i-й строки. В результате i-я строка нашей системы преобразуется к виду:

$$(b_i - a_i p_{i-1})x_i + c_i x_{i+1} = d_i - a_i q_{i-1}.$$

Чтобы получить единицу на главной диагонали матрицы, разделим i-ю строку на коэффициент $b_i - a_i p_{i-1}$. В результате для элементов p_i и q_i в окончательном виде i-й строки получаем следующие формулы:

$$\begin{aligned} p_i &= \frac{c_i}{b_i - a_i p_{i-1}}, \quad i = 2, 3, \dots, n-1, \quad p_1 = \frac{c_1}{b_1}, \\ q_i &= \frac{d_i - a_i q_{i-1}}{b_i - a_i p_{i-1}}, \quad i = 2, 3, \dots, n, \qquad q_1 = \frac{d_1}{b_1}. \end{aligned} \tag{2.46}$$

Продолжая исключение, получаем систему уравнений с двухдиагональной матрицей вида

$$\begin{pmatrix} 1 & p_1 & 0 & \dots & 0 \\ 0 & 1 & p_2 & \dots & 0 \\ \vdots & \ddots & \ddots & \ddots & \vdots \\ 0 & \dots & 0 & 1 & p_{n-1} \\ 0 & \dots & 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \\ \vdots \\ x_{n-1} \\ x_n \end{pmatrix} = \begin{pmatrix} q_1 \\ q_2 \\ \vdots \\ q_{n-1} \\ q_n \end{pmatrix}.$$

Это позволяет записать рекуррентные формулы:

$$\begin{aligned} x_n &= q_n, \\ x_i &= -p_i x_{i+1} + q_i, \quad i = n-1, n-2, \dots, 1, \end{aligned} \tag{2.47}$$

по которым находятся неизвестные x_i.

§ 2.14. Корректность и устойчивость метода прогонки

Рассмотрим вопросы корректности и устойчивости вычислений в рассмотренном выше методе трехточечной прогонки. Корректность означает выполнимость всех используемых при реализации прогонки формул, т. е. в данном случае не обращение в нуль знаменателей в формулах (2.46). Под устойчивостью понимается отсутствие прогрессивного накопления ошибок округления при выполнении операций умножения в формуле (2.47).

Для системы (2.45) с трехдиагональной матрицей условия строгого диагонального преобладания (2.44) принимают вид:

$$|b_i| > |a_i| + |c_i|, \quad i = 1, 2, \ldots, n, \tag{2.48}$$

где $a_1 = c_n = 0$.

Покажем, что при выполнении условий строгого диагонального преобладания (2.48) метод трехточечной прогонки (2.46), (2.47) корректен и устойчив. Согласно формулам (2.46) и неравенствам (2.48) получаем $|p_1| = |c_1|/|b_1| < 1$. Пусть по индукции $|p_j| < 1$, $j = 1, 2, \ldots, i-1$. Тогда, используя формулу (2.46), получаем

$$|p_i| = \frac{|c_i|}{|b_i - a_i p_{i-1}|} \le \frac{|c_i|}{|b_i| - |a_i|\,|p_{i-1}|} < \frac{|c_i|}{|b_i| - |a_i|} < \frac{|c_i|}{|c_i|} = 1,$$

т. е. $|p_i| < 1$ для всех i.

Так как

$$|b_i - a_i p_{i-1}| \ge |b_i| - |a_i||p_{i-1}| > |b_i| - |a_i| > 0, \quad i = 2, 3, \ldots, n-1,$$

то знаменатели в формулах (2.46) отличны от нуля. Это означает корректность метода прогонки.

Предположим, что при реальных вычислениях, решая систему (2.45) путем применения формул (2.46) и (2.47), получаем $\bar{x}_i = x_i + \varepsilon_i$ для $i = 1, 2, \ldots, n$, где ε_i – ошибка округления на i-м шаге. Тогда согласно формулам (2.47) получаем:

$$\bar{x}_i = -p_i \bar{x}_{i+1} + q_i, \quad i = n-1, \ldots, 1.$$

Вычитая из этого уравнения соотношения (2.47), находим:

$$\varepsilon_i = -p_i \varepsilon_{i+1}, \quad i = n-1, \ldots, 1,$$

откуда

$$|\varepsilon_i| = |p_i|\,|\varepsilon_{i+1}| < |\varepsilon_{i+1}|, \quad i = n-1, \ldots, 1,$$

т. е. вычисления по формуле (2.47) являются устойчивыми.

Было показано, что для интерполяционного кубического сплайна матрицы систем (2.38), (2.39), (2.41) и (2.43) для всех рассмотренных четырех типов краевых условий имеют строгое диагональное преобладание. Следовательно, системы (2.38), (2.39) и (2.43) могут быть устойчиво решены методом трехточечной прогонки (2.46), (2.47). Для решения системы (2.41) используется рассматриваемый ниже несколько более сложный вариант прогонки, который, однако, опять является модификацией метода исключения Гаусса.

§ 2.15. Метод фронтальной прогонки

Под методом *фронтальной* прогонки обычно понимается вариант метода прогонки, когда элементы матрицы и правой части системы линейных алгебраических уравнений заранее не вычисляются и не хранятся, а находятся непосредственно в процессе прогонки. При вычислении прогоночных коэффициентов на i-м шаге формируются элементы i-й «фронтальной» строки линейной системы, которые затем сразу же используются для нахождения i-х прогоночных коэффициентов. В процессе счета «фронт» постоянно движется. От обычной трехточечной прогонки метод фронтальной прогонки отличается большей экономичностью вычислений. Анализ корректности и устойчивости трехточечной прогонки остается в силе.

В качестве примера рассмотрим решение системы (2.38) методом фронтальной прогонки. Для экономии вычислений последнюю целесообразно переписать относительно неизвестных $\bar{M}_i = M_i/6$, $i = 0, 1, \ldots, N$. Применяя метод прогонки, вначале преобразуем исходную систему к системе с двухдиагональной верхней треугольной матрицей:

$$\begin{pmatrix} \alpha_0 & h_0 & 0 & 0 & \ldots & 0 \\ 0 & \alpha_1 & h_1 & 0 & \ldots & 0 \\ \vdots & \ddots & \ddots & \ddots & & \vdots \\ 0 & \ldots & 0 & 0 & \alpha_{N-1} & h_{N-1} \\ 0 & \ldots & 0 & 0 & 0 & \alpha_N \end{pmatrix} \begin{pmatrix} \bar{M}_0 \\ \bar{M}_1 \\ \vdots \\ \bar{M}_{N-1} \\ \bar{M}_N \end{pmatrix} = \begin{pmatrix} \beta_0 \\ \beta_1 \\ \vdots \\ \beta_{N-1} \\ \beta_N \end{pmatrix},$$

где диагональные элементы α_i вычисляются по формулам:

$$\begin{aligned} \alpha_0 &= 2h_0; \\ \alpha_i &= 2(h_{i-1} + h_i) - \frac{h_{i-1}^2}{\alpha_{i-1}}; \quad i = 1, 2, \ldots, N-1; \\ \alpha_N &= 2h_{N-1} - \frac{h_{N-1}^2}{\alpha_{N-1}}; \end{aligned}$$

а элементы правой части β_i находятся согласно рекуррентным соотношениям:

$$\beta_0 = f[x_0, x_1] - f'_0;$$

$$\begin{aligned}\beta_i &= \delta_i f - \frac{h_{i-1}\beta_{i-1}}{\alpha_{i-1}}; \quad i = 1, 2, \ldots, N-1;\\ \beta_N &= f'_N - f[x_{N-1}, x_N] - \frac{h_{N-1}\beta_{N-1}}{\alpha_{N-1}}.\end{aligned}$$

Окончательно коэффициенты $\bar{M}_i$ вычисляются по формулам:

$$\begin{aligned}\bar{M}_N &= \frac{\beta_N}{\alpha_N};\\ \bar{M}_i &= \frac{\beta_i - h_i\bar{M}_{i+1}}{\alpha_i}; \quad i = N-1, \ldots, 0.\end{aligned}$$

Приведенный алгоритм весьма экономичен. В частности, единицы на главной диагонали выписанной выше двухдиагональной матрицы не формируются из соображений уменьшения числа делений.

Значения сплайна S на отрезке $[x_i, x_{i+1}]$ находятся по формуле (2.35). При их многократном вычислении целесообразно переписать эту формулу в виде:

$$S(x) = f_i + (x - x_i)(b_i + (x - x_i)(c_i + (x - x_i)d_i)),$$

где

$$\begin{aligned}b_i &= f[x_i, x_{i+1}] - h_i(2\bar{M}_i + \bar{M}_{i+1});\\ c_i &= 3\bar{M}_i;\\ d_i &= (\bar{M}_{i+1} - \bar{M}_i)/h_i.\end{aligned}$$

Это позволяет уменьшить число выполняемых арифметических операций.

§ 2.16. Пример построения кубического сплайна

Пример 2.5. Кубический сплайн S интерполирует данные табл. 2.3. Найти значение $S(0,5)$, если $S'(0) = S'(1) = 0$. Использовать метод фронтальной прогонки. Начертить график сплайна S.

Таблица 2.3

i	0	1	2	3
x_i	0	1/3	2/3	1
f_i	1	0	0	0

Решение. Данные табл. 2.3 являются равноотстоящими: $h_i = h$ для всех i. Для экономии вычислений обозначим $\tilde{M}_i = S''(x_i)h^2/6$, $i = 0, 1, 2, 3$. Формулу (2.35) для интерполяционного кубического сплайна на отрезке $[x_i, x_{i+1}]$ можно переписать в виде:

$$S(x) = f_i(1-t) + f_{i+1}t - t(1-t)\big[(2-t)\tilde{M}_i + (1+t)\tilde{M}_{i+1}\big], \qquad (2.49)$$

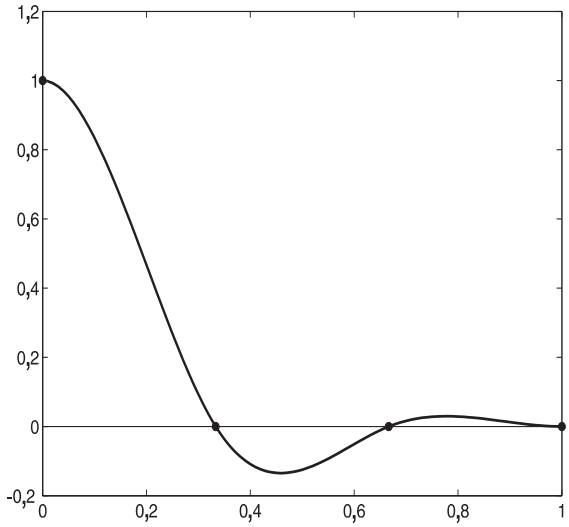


Рис. 2.4. График кубического сплайна S, интерполирующего данные табл. 2.3 и удовлетворяющего краевым условиям $S'(0) = S'(1) = 0$.

где $t = (x - x_i)/h$.

Условия гладкости (2.37) дают

$$\tilde{M}_{i-1} + 4\tilde{M}_i + \tilde{M}_{i+1} = h\delta_i f, \quad i = 1, 2.$$

Краевые соотношения $S'(0) = S'(1) = 0$ эквивалентны уравнениям:

$$2\tilde{M}_0 + \tilde{M}_1 = hf[x_0, x_1], \quad \tilde{M}_2 + 2\tilde{M}_3 = hf[x_2, x_3].$$

Таким образом, приходим к системе линейных уравнений:

$$\begin{pmatrix} 2 & 1 & 0 & 0 \\ 1 & 4 & 1 & 0 \\ 0 & 1 & 4 & 1 \\ 0 & 0 & 1 & 2 \end{pmatrix} \begin{pmatrix} \tilde{M}_0 \\ \tilde{M}_1 \\ \tilde{M}_2 \\ \tilde{M}_3 \end{pmatrix} = \begin{pmatrix} hf[x_0, x_1] \\ h\delta_1 f \\ h\delta_2 f \\ hf[x_2, x_3] \end{pmatrix} \tag{2.50}$$

Отметим, что уравнения этой системы получаются из системы (2.38) умножением неизвестных на масштабирующий множитель $h^2/6$.

Для решения системы (2.50) применяем метод фронтальной прогонки. Прогоночные коэффициенты находим по формулам:

$$\alpha_0 = 2, \qquad \beta_0 = hf[x_0, x_1],$$

$$\alpha_i = 4 - \frac{1}{\alpha_{i-1}}, \quad i = 1, 2, \qquad \beta_i = h\delta_i f - \frac{\beta_{i-1}}{\alpha_{i-1}}, \quad i = 1, 2,$$

$$\alpha_3 = 2 - \frac{1}{\alpha_2}, \qquad \beta_3 = hf[x_2, x_3] - \frac{\beta_2}{\alpha_2}.$$

Это позволяет преобразовать систему (2.50) к виду:

$$\begin{pmatrix} \alpha_0 & 1 & 0 & 0 \\ 0 & \alpha_1 & 1 & 0 \\ 0 & 0 & \alpha_2 & 1 \\ 0 & 0 & 0 & \alpha_3 \end{pmatrix} \begin{pmatrix} \tilde{M}_0 \\ \tilde{M}_1 \\ \tilde{M}_2 \\ \tilde{M}_3 \end{pmatrix} = \begin{pmatrix} \beta_0 \\ \beta_1 \\ \beta_2 \\ \beta_3 \end{pmatrix}.$$

Используя данные табл. 2.3, имеем $hf[x_0, x_1] = -1$, $f[x_1, x_2] = 0$, $f[x_2, x_3] = 0$. Вычисления дают следующие значения прогоночных коэффициентов:

$$\begin{array}{llll} \alpha_0 = 2, & \alpha_1 = \frac{7}{2}, & \alpha_2 = \frac{26}{7}, & \alpha_3 = \frac{45}{26}, \\ \beta_0 = -1, & \beta_1 = \frac{3}{2}, & \beta_2 = -\frac{3}{7}, & \beta_3 = \frac{3}{26}. \end{array}$$

Неизвестные $\tilde{M}_i$ находим по формулам:

$$\begin{aligned} \tilde{M}_3 &= \frac{\beta_3}{\alpha_3}, \\ \tilde{M}_i &= \frac{\beta_i - \tilde{M}_{i+1}}{\alpha_i}, \quad i = 2, 1, 0. \end{aligned}$$

Рекуррентным счетом получаем следующие значения неизвестных $\tilde{M}_i$:

$$\tilde{M}_3 = \tfrac{1}{15}, \quad \tilde{M}_2 = -\tfrac{2}{15}, \quad \tilde{M}_1 = \tfrac{7}{15}, \quad \tilde{M}_0 = -\tfrac{11}{15}.$$

Теперь по формуле (2.49) вычисляем

$$S(0,5) = \frac{f_1 + f_2}{2} - \frac{3}{8}(\tilde{M}_1 + \tilde{M}_2) = -\frac{3}{8}\left(\frac{7}{15} - \frac{2}{15}\right) = -\frac{1}{8}.$$

График кубического сплайна S приведен на рис. 2.4.

§ 2.17. Инвариантность интерполяционных кубических сплайнов

Рассмотрим преобразование вещественной оси $\mathbb{R} \to \bar{\mathbb{R}}$ вида $\bar{x} = px + q$, где p, q – вещественные числа и $p \neq 0$. Это преобразование осуществляет сдвиг и растяжение (сжатие) на вещественной оси $\mathbb{R}$. В частности, сетка $a = x_0 < \ldots < x_N = b$ преобразуется в сетку $\{\ \bar{x}_i \mid \bar{x}_i = px_i + q,\ i = 0, 1, \ldots, N\ \}$.

Покажем, что интерполяционный кубический сплайн S инвариантен относительно линейных преобразований, т. е. его значения после применения линейного преобразования не изменяются: $S(x) = \bar{S}(\bar{x})$.

Формула (2.35) при подстановке $x = (\bar{x} - q)/p$ преобразуется к виду:

$$S_i(x) = \bar{M}_i \frac{(\bar{x}_{i+1} - \bar{x})^3}{6\bar{h}_i} + \bar{M}_{i+1} \frac{(\bar{x} - \bar{x}_i)^3}{6\bar{h}_i} + \left(f_i - \bar{M}_i \frac{\bar{h}_i^2}{6}\right) \frac{\bar{x}_{i+1} - \bar{x}}{\bar{h}_i} +$$

$$+\Big(f_{i+1} - \bar{M}_{i+1}\frac{\bar{h}_i^2}{6}\Big)\frac{\bar{x}-\bar{x}_i}{\bar{h}_i} = \bar{S}_i(\bar{x}),$$

где $\bar{h}_i = ph_i$, $\bar{M}_j = p^{-2}M_j$, $j = i, i+1$.

Для условий гладкости (2.37) имеем:

$$\begin{aligned}\bar{h}_{i-1}\bar{M}_{i-1} + 2(\bar{h}_{i-1}+\bar{h}_i)\bar{M}_i + \bar{h}_i\bar{M}_{i+1} &= 6(f[\bar{x}_i,\bar{x}_{i+1}] - f[\bar{x}_{i-1},\bar{x}_i]),\\ &\qquad i = 1,2,\ldots,N-1.\end{aligned}$$

Краевые условия принимают следующий вид:

1. $2\bar{M}_0 + \bar{M}_1 = \dfrac{6}{\bar{h}_0}\Big(f[\bar{x}_0,\bar{x}_1] - \dfrac{1}{p}f'_0\Big)$,

   $\bar{M}_{N-1} + 2\bar{M}_N = \dfrac{6}{\bar{h}_{N-1}}\Big(\dfrac{1}{p}f'_N - f[\bar{x}_{N-1},\bar{x}_N]\Big)$;
2. $\bar{M}_0 = \dfrac{1}{p^2}f''_0$ и $\bar{M}_N = \dfrac{1}{p^2}f''_N$;
3. $f_{N+i} = f_i$, $\bar{M}_{N+i} = \bar{M}_i$, $\bar{h}_{N+i} = \bar{h}_i$ для всех i;
4. $\dfrac{\bar{M}_{i+1}-\bar{M}_i}{\bar{h}_i} = \dfrac{\bar{M}_i - \bar{M}_{i-1}}{\bar{h}_{i-1}}$, $i = 1, N-1$.

Так как

$$\frac{d}{dx}\bar{S}(\bar{x}) = \frac{d}{d\bar{x}}\bar{S}(\bar{x})\frac{d\bar{x}}{dx} = p\bar{S}'(\bar{x}) \quad \text{и} \quad \frac{d^2}{dx^2}\bar{S}(\bar{x}) = p^2\bar{S}''(\bar{x}),$$

то полученные соотношения доказывают инвариантность интерполяционных кубических сплайнов относительно линейных преобразований.

§ 2.18. Аппроксимация кубическими B-сплайнами

Описанные интерполяционные кубические сплайны не всегда удобны в приложениях. Для их построения требуется решать систему линейных уравнений и если нам нужно исправить даже одно значение, то эту систему приходится решать заново. Более удобны *локально-аппроксимационные сплайны*, основанные на представлении сплайна в виде линейной комбинации базисных сплайнов (сокращенно B-сплайнов). Ограничимся рассмотрением кубических локально-аппроксимационных сплайнов.

Как было показано в разделе 2.9, множество кубических сплайнов, удовлетворяющих определению 2.1, образует линейное пространство $S_3(\Delta)$ размерности $N+3$. Построим в этом пространстве базис из функций с конечными носителями минимальной длины. Расширим сетку Δ точками $x_{-3} < x_{-2} < x_{-1} < a$ и $b < x_{N+1} < x_{N+2} < x_{N+3}$ и введем в рассмотрение кубические B-сплайны:

$$B_{j,3}(x) = (x_{j+n} - x_j)\varphi_3[x; x_j, \ldots, x_{j+4}], \quad j = -3, -2, \ldots, N-1,$$

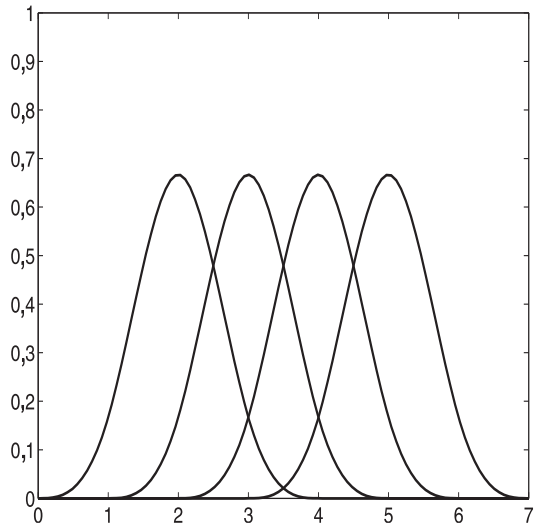


Рис. 2.5. Кубические базисные сплайны $B_{j,4}$ на целочисленной сетке.

где $\varphi_3(x,y) = (x-y)_+^3 = [\max(0, x-y)]^3$. Вид функций $B_{j,4}$ на целочисленной сетке Δ приведен на рис. 2.5.

Пользуясь формулой (2.9), сплайн $B_{j,3}$ можно еще переписать в виде:

$$B_{j,3}(x) = (x_{j+4} - x_j) \sum_{k=j}^{j+4} \frac{(x-x_k)_+^3}{\omega'_{4,j}(x_k)}, \quad \omega_{4,j}(x) = \prod_{k=j}^{j+4}(x-x_k).$$

Поступая, как и при доказательстве теоремы 2.6, нетрудно показать, что функции $B_{j,3}$, $j = -3, \ldots, N-1$ линейно независимы на $[a,b]$ и образуют базис в пространстве $S_3(\Delta)$. Они обладают следующими свойствами: $B_{j,3}(x) > 0$ для $x \in (x_j, x_{j+4})$ и $B_{j,3}(x) \equiv 0$ в противном случае,

$$(y-x)^3 = \sum_{j=-3}^{N-1} (y-x_{j+1})(y-x_{j+2})(y-x_{j+3})B_{j,3}(x), \quad x \in [a,b].$$

Это равенство может быть также переписано в эквивалентном виде:

$$x^\alpha = \frac{1}{C_3^\alpha} \sum_{j=-3}^{N-1} sym_\alpha(x_{j+1}, x_{j+2}, x_{j+3})B_{j,3}(x), \quad x \in [a,b], \tag{2.51}$$

где $C_3^\alpha = \binom{3}{\alpha}$ – обычный биномиальный коэффициент и

$$\begin{aligned} sym_0(x,y,z) &= 1, \\ sym_1(x,y,z) &= x+y+z, \\ sym_2(x,y,z) &= xy+xz+yz, \\ sym_3(x,y,z) &= xyz. \end{aligned}$$

Для вычисления значений базисных сплайнов $B_{j,3}$ используется рекуррентная формула (см. [17])

$$B_{j,k}(x) = \frac{x - x_j}{x_{j+k-1} - x_j} B_{j,k-1}(x) + \frac{x_{j+k} - x}{x_{j+k} - x_{j+1}} B_{j+1,k-1}(x), \quad k \geq 2, \tag{2.52}$$

где

$$B_{j,0}(x) = \begin{cases} 1, & \text{если } x \in [x_j, x_{j+1}); \\ 0 & \text{в противном случае.} \end{cases}$$

Рассмотрим следующую формулу локальной аппроксимации кубическими базисными сплайнами

$$S_f(x) = \sum_{j=-3}^{N-1} b_{j+2} B_{j,3}(x). \tag{2.53}$$

Здесь можно положить $b_j = f_j$, если данные неточные и требуется сглаживание погрешностей и $b_j = b_{j,-1} f_{j-1} + b_{j,0} f_j + b_{j,1} f_{j+1}$ при малых погрешностях. В последнем случае, если взять $b_{j,0} = 1 - b_{j,-1} - b_{j,1}$,

$$b_{j,-1} = -\frac{h_j^2}{3h_{j-1}(h_{j-1} + h_j)}, \quad b_{j,1} = -\frac{h_{j-1}^2}{3h_j(h_{j-1} + h_j)},$$

то формула (2.53) будет точна на кубических многочленах. Для этого достаточно убедиться, что она точна на мономах 1, x, x^2, x^3. Подставляя последние в формулу (2.53), получаем равенства (2.51).

Согласно формуле (2.14), имеем:

$$f(x) = L_{i,3}(x) + R_{i,3}(x),$$

где $R_{i,3}(x) = f[x_{i-1}, \ldots, x_{i+2}, x]\omega_{i-1,3}(x)$.

Так как сплайн S_f точен на кубических многочленах, то

$$S_f(x) = L_{i,3}(x) + S_{R_{i,3}}(x).$$

Согласно формуле (2.53), на отрезке $[x_i, x_{i+1}]$ имеем:

$$\begin{aligned} S_{R_{i,3}}(x) \;\; = \;\; & b_{i-1,-1} R_{i,3}(x_{i-2}) B_{i-3,3}(x) + b_{i+2,1} R_{i,3}(x_{i+3}) B_{i,3}(x) + \\ & + \sum_{j=i-1}^{i+2} \psi_j(x) R_{i,3}(x_j), \end{aligned}$$

где ψ_j – некоторые кубические многочлены. Так как остаточный член $R_{i,3}(x_j) = 0$ для $j = i-1, \ldots, i+2$, то

$$S_f(x) = L_{i,3}(x) + b_{i-1,-1} R_{i,3}(x_{i-2}) B_{i-3,3}(x) + b_{i+2,1} R_{i,3}(x_{i+3}) B_{i,3}(x).$$

Подставляя сюда выражения для B-сплайнов и остаточного члена $R_{i,3}$, получаем опять формулу (2.31).

§ 2.19. Задачи

2.1. Пусть имеется набор исходных данных (x_i, f_i), $i = 0, 1, \ldots, N$ таких, что их $(n+1)$-е разделенные разности равны нулю. Покажите, что тогда разделенные разности порядков от $n+2$ до N тоже будут равны нулю.

2.2. Пусть $L_{i,n}$ – многочлен Лагранжа степени n, интерполирующий данные (x_j, f_j), $j = i, \ldots, i+n$. Докажите, что имеет место равенство

$$L_{i+1,n}(x) = L_{i,n}(x) + (x_{i+n+1} - x_i) f[x_i, \ldots, x_{i+n+1}](x - x_{i+1}) \ldots (x - x_{i+n}). \quad (2.54)$$

2.3. Пусть имеется набор точек (x_i, f_i), $i = 0, 1, \ldots, N$ таких, что разделенные разности порядка $n+1$ $(n < N)$ равны нулю. Покажите, что тогда существует многочлен L_n степени n такой, что $L_n(x_i) = f_i$ для $i = 0, 1, \ldots, N$.

2.4. Покажите, что интерполяционный многочлен Лагранжа L_N может быть записан в виде

$$L_N(x) = \frac{\sum_{j=0}^{N} [w_j f_j]/(x - x_j)}{\sum_{j=0}^{N} w_j/(x - x_j)},$$

где $w_j = 1/\omega'_N(x_j)$. Данная формула обычно называется *барицентрическим* представлением интерполяционного многочлена Лагранжа.

2.5. Функция $f(x) = \cos(\pi x/2)$ задана табл. 2.4 своих значений.

Таблица 2.4

i	x_i	f_i
0	0,0	1,0
1	1,0	0,0
2	2,0	-1,0
3	3,0	0,0

а) постройте таблицу разделенных разностей;

б) запишите интерполяционный многочлен Ньютона L_3;

в) по схеме Горнера найдите значения $L_3(1{,}5)$ и $L'_3(1{,}5)$;

г) сравните полученные значения с точными: $f(1{,}5)$ и $f'(1{,}5)$.

2.6. Многочлен четвертой степени P_4 удовлетворяет условиям:

$$\Delta^4 P_4(0) = 24, \quad \Delta^3 P_4(0) = 6, \quad \Delta^2 P_4(0) = 0,$$

где $\Delta P_4(x) = P_4(x+1) - P_4(x)$. Найдите $\Delta^2 P_4(10)$.

2.7. Рассмотрите функцию $B_{j,1}$, определенную формулой

$$B_{j,1}(x) = (x_{j+2} - x_j)\varphi[x; x_j, x_{j+1}, x_{j+2}],$$

где $\varphi(x,y) = (x-y)_+ = \max(0, x-y)$, или, в силу формулы (2.9):

$$\begin{aligned} B_{j,1}(x) &= (x_{j+2} - x_j) \sum_{k=j}^{j+2} \frac{(x-x_k)_+}{\omega'_{j,2}(x_k)}, \\ \omega_{j,2}(x) &= (x-x_j)(x-x_{j+1})(x-x_{j+2}). \end{aligned}$$

Покажите, что функции $B_{j,1}$, $j = -1, 0, \ldots, N-1$ линейно независимы на $[a,b]$ и образуют базис в пространстве ломаных $S_1(\Delta)$ (см. § 2.6).

2.8. Функция $f(x) = \sin \pi x$ приближается на отрезке $[0,1]$ с помощью кусочно-линейной интерполяции по точкам $x_i = i/N$, $i = 0, 1, \ldots, N$. Какова точность приближения при $N = 5$? Сколько узлов интерполяции требуется для достижения точности приближения $\varepsilon = 10^{-4}$?

2.9. Функция $f(x) = sin^2(x)$ приближается на отрезке $[0, \pi]$ с помощью кусочно-линейной интерполяции по точкам $x_i = i\pi/N$, $i = 0, 1, \ldots, N$. Найдите оптимальный шаг численного дифференцирования $h = \pi/N$, если значения f вычисляются с точностью $\varepsilon = 10^{-2}$.

2.10. Покажите, что функция

$$B_3^L(x) = \begin{cases} \varphi(2 - |x|), & 1 \le |x| < 2; \\ 1 - |x| + \varphi(|x|) - 2\varphi(1 - |x|), & 0 \le x < 1; \\ 0 \qquad\qquad \text{в противном случае,} \end{cases}$$

где $\varphi(x) = (x+1)x(x-1)/6$, является кубическим лагранжевым базисным сплайном с узлами $\{-2, -1, 0, 1, 2\}$. Сформулируйте свойства, которыми должна обладать функция B_3^L, чтобы данное утверждение было верно. В частности, проверьте выполнение тождества

$$\sum_j B_3^L(x - j) \equiv 1 \quad \text{для} \quad x \in (-\infty, \infty).$$

2.11. Функция $f(x) = \exp(-x)$ приближается на отрезке $[0,3]$ кубическим лагранжевым сплайном с точками интерполяции $x_i = 3i/N$, $i = 0, 1, \ldots, N$. Сколько точек интерполяции требуется взять, чтобы обепечить точность приближения $\varepsilon = 10^{-6}$? Какова будет точность приближения при $N = 5$?

2.12. Рассмотрите функцию B_3, определенную формулой:

$$B_3(x) = \begin{cases} \dfrac{2}{3} - x^2 + \dfrac{1}{2}|x|^3 & \text{при} \quad |x| < 1, \\ \dfrac{1}{6}(2 - |x|)^3 & \text{при} \quad 1 \le |x| < 2, \\ 0 & \text{при} \quad 2 \le |x|. \end{cases}$$

Покажите, что B_3 – дважды непрерывно дифференцируемая функция, обладающая свойствами:

$$B_3(x) > 0 \quad \text{для} \quad x \in (-2, 2),$$

$$\sum_j B_3(x - j) \equiv 1 \quad \text{для} \quad x \in (-\infty, \infty).$$

2.13. Кубический сплайн S интерполирует данные табл. 2.5. Найдите значение $S(0{,}5)$, если $S'(0) = S'(1) = 0$. Начертите график сплайна S.

Таблица 2.5

i	0	1	2	3
x_i	0	1/3	2/3	1
f_i	0	1	0	0

2.14. Рассмотрите функцию S, определенную формулой:

$$S(x) = \begin{cases} 1 - 2x, & x < -3; \\ 28 + 25x + 9x^2 + x^3, & -3 \le x < -1; \\ 26 + 19x + 3x^2 - x^3, & -1 \le x < 0; \\ 26 + 19x + 3x^2 - 2x^3, & 0 \le x < 3; \\ -163 + 208x - 60x^2 + 5x^3, & 3 \le x < 4; \\ 157 - 32x, & 4 \le x. \end{cases}$$

Покажите, что функция S – кубический сплайн с узлами $\{-3, -1, 0, 3, 4\}$. Сформулируйте свойства S, необходимые для справедливости этого утверждения.

2.15. Рассмотрите функцию S, определенную формулой:

$$S(x) = \begin{cases} (x-2)^3 + a(x-1)^2, & x \in (-\infty, 2]; \\ (x-2)^3 - (x-3)^2, & x \in [2, 3]; \\ (x-3)^3 + b(x-2)^2, & x \in [3, +\infty). \end{cases}$$

Можно ли подобрать коэффициенты a и b таким образом, чтобы функция S являлась кубическим сплайном?

2.16. Кубический сплайн S интерполирует следующие данные $\{x_i\} = \{0, 1, 2, 3\}$, $\{f_i\} = \{1, 1, 0, 10\}$ и удовлетворяет краевым условиям $S''(0) = S''(3) = 0$. Совпадает ли он с функцией f, определенной формулой:

$$f(x) = \begin{cases} 1 + x - x^3, & x \in [0, 1]; \\ 1 - 2(x-1) - 3(x-1)^2 + 4(x-1)^3, & x \in [1, 2]; \\ 4(x-2) + 9(x-2)^2 - 3(x-2)^3, & x \in [2, 3]. \end{cases}$$

2.17. Покажите, что всякий кубический сплайн $S \in C^2$ с узлами в точках x_i, $i = 1, 2, \ldots, N-1$ допускает однозначное представление в виде

$$S(x) = P_3(x) + \sum_{i=1}^{N-1} C_i (x - x_i)_+^3 / 6, \quad E_+ = \max(E, 0),$$

где $C_i = S'''(x_i + 0) - S'''(x_i - 0)$, P_3 – некоторый кубический многочлен.

2.18. Пусть $P_{j,3}$, $j = i, i+1$, – два кубических многочлена, удовлетворяющих условиям интерполяции $P_{i,3}(x_j) = f_j$, $P_{i+1,3}(x_{j+1}) = f_{j+1}$ для $j = i, \ldots, i+3$. Покажите справедливость формулы

$$P_{i+1,3}(x) = P_{i,3}(x) + f[x_i, \ldots, x_{i+4}](x_{i+4} - x_i)(x - x_{i+1})(x - x_{i+2})(x - x_{i+3}).$$

2.19. Пусть задана последовательность точек (x_i, f_i), $i = 0, 1, \ldots, N$ таких, что $f[x_i, \ldots, x_{i+4}] = 0$ для $i = 0, 1, \ldots, N-4$. Докажите, что существует интерполяционный кубический многочлен P_3 такой, что $P_3(x_i) = f_i$, $i = 0, 1, \ldots, N$.
Указание. Воспользуйтесь решением задачи 2.18.

2.20. В евклидовой плоскости на окружности единичного радиуса заданы три точки: $P_0 = (1, 0)$, $P_1 = (-\frac{1}{2}, \frac{\sqrt{3}}{2})$, $P_2 = (-\frac{1}{2}, -\frac{\sqrt{3}}{2})$. Постройте периодический параметрический кубический сплайн $S(t) = (S_x(t), S_y(t))$ такой, что $S(t_i) = P_i$, $i = 0, 1, 2$ и $S(t_3) = P_0$. Найдите точку пересечения графика сплайна с осью x при $x < 0$. Используйте параметризацию по суммарной длине хорд:

$$t_0 = 0, \quad t_i = t_{i-1} + \frac{|P_i - P_{i-1}|}{\sum_{j=1}^{3} |P_j - P_{j-1}|}, \quad i = 1, 2, 3,$$

где $|\cdot|$ обозначает евклидово расстояние. Начертите графики сплайнов S_x, S_y и S. Изменится ли график сплайна при переходе к равномерной параметризации:

$$t_0 = 0; \quad t_i = t_{i-1} + h; \quad i = 1, 2, 3; \quad h = 1/3?$$

2.21. Пользуясь рекуррентной формулой (2.52), найдите явное выражение для квадратического базисного сплайна $B_{j,3}$. Покажите, что

$$B_{j,3}(x_{j+1}) = \frac{x_{j+1} - x_j}{x_{j+2} - x_j} \quad \text{и} \quad B_{j,3}(x_{j+2}) = \frac{x_{j+3} - x_{j+2}}{x_{j+3} - x_{j+1}}.$$

Какой гладкостью обладает сплайн $B_{j,3}$? Докажите тождество

$$(y - x)^2 = \sum_{j=-2}^{N-1} (y - x_{j+1})(y - x_{j+2}) B_{j,3}(x) \quad \text{для} \quad x \in [a, b].$$

Нарисуйте график сплайна $B_{j,3}$ в случае кратных узлов, когда $x_{j+2} = x_{j+3}$ и $x_{j+1} = x_{j+2} = x_{j+3}$.

Глава 3
Метод наименьших квадратов и сплайн-сглаживание

Очень часто возникает необходимость выразить в виде функциональной зависимости связь между величинами, которые заданы в виде набора точек с координатами (x_i, y_i), $i = 0, 1, \ldots, N$. Если необходимо использовать эти данные для вычислений на компьютере, то сразу появляются следующие проблемы:

1. в значениях y_i наверняка имеются погрешности эксперимента; было бы желательно каким-либо образом «сгладить» те отклонения, которые обусловлены ошибками эксперимента;

2. может оказаться желательным знать значения $\tilde{y}$, соответствующие промежуточным значениям $\tilde{x}$;

3. может оказаться, что необходимо экстраполировать функциональную зависимость, т. е. найти значение y, соответствующее значению x, лежащему вне области эксперимента (иногда это является главной целью эксперимента и вычислений); в частности, это относится к экономической информации.

Все эти соображения приводят нас к выводу, что желательно было бы установить некоторую функциональную зависимость между x и y в виде по возможности простой формулы. Вопрос состоит в том, как найти кривую, которая *приближенно* соответствует исходной информации с достаточной точностью. Таким образом, нужно выработать критерий, согласно которому та или иная кривая является достаточно «хорошим» приближением к исходной информации.

§3.1. Критерий наименьших квадратов

Введем понятие *отклонения* экспериментальной точки как разность между экспериментальной ординатой y_i и той, которая вычислена из функциональной зависимости. Вопрос о том, является ли кривая достаточно «хорошим» приближением к экспериментальным данным, можно поставить в следующем виде: какое

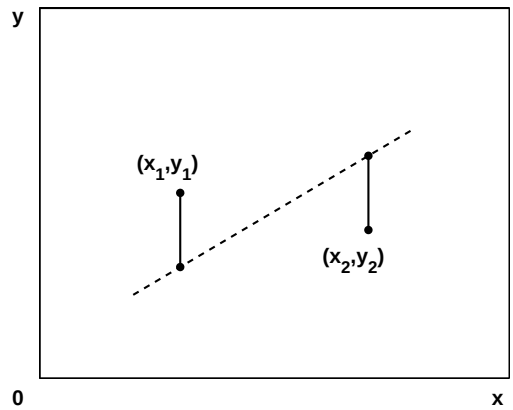


Рис. 3.1. Пример, показывающий, что сумма отклонений не может служить критерием для подбора функциональной зависимости.

условие необходимо наложить на отклонение экспериментальных точек от кривой, чтобы эта кривая представляла экспериментальные данные с достаточной точностью?

Казалось бы, что наиболее простое и логичное условие состоит в том, чтобы сумма отклонений точек от кривой была наименьшей. Если обозначить через $\tilde{y}$ значение y, вычисленное из функциональной зависимости, то это условие можно записать так: требуется, чтобы сумма отклонений

$$\sum_{i=0}^{N}(y_i - \tilde{y}_i)$$

была минимальной. Привлекательность этого простого критерия однако сразу становится сомнительной, стоит только рассмотреть простую задачу о проведении прямой линии через две точки, как это показано на рис. 3.1. Мы видим, что штриховая линия удовлетворяет нашему критерию, но эту линию никак нельзя признать удовлетворительным приближением к экспериментальным данным. Можно попытаться обойти это затруднение, используя в критерии сумму абсолютных значений отклонений, т. е. требуя, чтобы величина

$$\sum_{i=0}^{N}|y_i - \tilde{y}_i|$$

стала минимальной. Но в этом случае для нахождения минимума нельзя воспользоваться производной, так как абсолютное значение не имеет производной в точке минимума. Можно было бы наложить условие, согласно которому максимальное отклонение должно стать наименьшим (приближение Чебышева), но для опре-

деления функциональной зависимости на основе этого критерия приходится использовать длинную и сложную итерационную процедуру.

Поэтому в данном случае мы воспользуемся критерием наименьших квадратов, т. е. будем искать такую функциональную зависимость, при которой сумма квадратов отклонений

$$\sum_{i=0}^{N}(y_i - \tilde{y}_i)^2$$

обращается в минимум. Это выражение, как мы увидим ниже, можно продифференцировать для нахождения минимума. Такой критерий во многих практических случаях приводит к линейным уравнениям, которые легко решить, по крайней мере в принципе.

Наконец, можно статистически обосновать, что критерий наименьших квадратов дает достаточно хорошее приближение функциональной зависимости к экспериментальным данным, даже если отвлечься от вопроса о практике вычислений.

§3.2. Нормальная система метода наименьших квадратов

Рассмотрим теперь вопрос о том, как при использовании критерия наименьших квадратов получается система уравнений для определения функциональной зависимости y от x.

Будем считать, что имеются исходные данные (x_i, y_i), $i = 0, 1, \ldots, N$, где абсциссы упорядочены по возрастанию $a = x_0 < x_1 < \ldots < x_N = b$ и задана система линейно независимых на отрезке $[a, b]$ функций φ_j, $j = 1, 2, \ldots, M$ $(M \ll N)$. Для приближенного описания исходных данных рассмотрим функцию $S(x) = \sum_{j=1}^{M} c_j\varphi_j(x)$ такую, что среднеквадратическое отклонение

$$E_M \equiv E_M(c_1, c_2, \ldots, c_M) = \sum_{i=0}^{N} p_i(S(x_i) - y_i)^2 = \sum_{i=0}^{N} p_i\Big(\sum_{j=1}^{M} c_j\varphi_j(x_i) - y_i\Big)^2 \qquad (3.1)$$

достигает минимума. Здесь величины $p_i > 0$, называемые весами, обычно выбирают из соображений точности задания y_i.

Необходимое условие экстремума $\partial E_M/\partial c_k = 0$, $k = 1, 2, \ldots, M$ дает нам систему M линейных алгебраических уравнений для нахождения M искомых коэффициентов c_k:

$$\sum_{j=1}^{M} c_j\Big(\sum_{i=0}^{N} p_i\varphi_j(x_i)\varphi_k(x_i)\Big) = \sum_{i=0}^{N} p_i y_i\varphi_k(x_i), \quad k = 1, 2, \ldots, M, \qquad (3.2)$$

называемую *нормальной системой метода наименьших квадратов* (МНК). Так как функции φ_j, $j = 1, 2, \ldots, M$ по условию линейно независимы на отрезке $[a, b]$, то можно показать, что определитель нормальной системы отличен от нуля и ее решение может быть получено, например, методом исключения Гаусса или одним из итерационных методов (гл. 5).

Далее для простоты изложения будем полагать $p_i = 1$ для всех i. Это предположение является естественным, если нет дополнительной информации об исходных данных.

§3.3. Приближение многочленами

В качестве системы функций $\{\varphi_j\}$ могут быть взяты мономы $\{x^{j-1}\}$ и в этом случае функция $S(x) = \sum_{j=1}^{M} c_j x^{j-1}$ является многочленом степени $M - 1$. Нормальные уравнения для нахождения коэффициентов многочлена степени $M - 1$ можно записать в виде:

$$\begin{aligned}
(N+1)c_1 + \sum x_i c_2 + \ldots + \sum x_i^{M-1} c_M &= \sum y_i, \\
\sum x_i c_1 + \sum x_i^2 c_2 + \ldots + \sum x_i^M c_M &= \sum x_i y_i, \\
\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots &\ldots \quad \ldots\ldots \\
\sum x_i^{M-1} c_1 + \sum x_i^M c_2 + \ldots + \sum x_i^{2M-2} c_M &= \sum x_i^{M-1} y_i,
\end{aligned}$$

где $\sum = \sum_{i=0}^{N}$.

Для многочлена степени $M-1$ получается система M уравнений относительно M коэффициентов c_k. Количество сумм, которые необходимо вычислить, равно $3M - 1$.

Если исходные данные y_i являются значениями некоторого многочлена P_l степени $l \le M - 1$, т. е. $y_i = P_l(x_i)$, $i = 0, 1, \ldots, N$, то МНК будет воспроизводить этот многочлен, т. е. $S(x) \equiv P_l(x)$. Это свойство полезно использовать для тестирования программы вычисления многочлена наилучшего среднеквадратического приближения. Отметим, что при $M = N + 1$ построенный по МНК многочлен будет интерполировать исходные данные и совпадает с многочленом Лагранжа степени N.

В частном случае многочлена первой степени $S(x) = c_1 + c_2 x$ получаем

$$(N+1)c_1 + \sum_{i=0}^{N} x_i c_2 = \sum_{i=0}^{N} y_i, \quad \sum_{i=0}^{N} x_i c_1 + \sum_{i=0}^{N} x_i^2 c_2 = \sum_{i=0}^{N} x_i y_i.$$

Решение этой системы имеет вид:

$$c_1 = \frac{1}{N+1}\Big(\Sigma y_i - c_2 \Sigma x_i\Big), \quad c_2 = \frac{\sum x_i y_i - \frac{1}{N+1}\sum x_i \sum y_i}{\sum x_i^2 - \frac{1}{N+1}\Big(\sum x_i\Big)^2}, \quad \Sigma = \Sigma_{i=0}^{N}.$$

Пример 3.1. В табл. 3.1 приведены экспериментальные данные из [19] по зависимости теплоемкости воды c_p от температуры T, причем теплоемкость при 15^oC принята за единицу.

Построив по МНК кубический многочлен, получим

$$S(x) = 1{,}006447 - 0{,}0004987884T + 0{,}000008459123T^2 - 0{,}0000000345339T^3.$$

На рис. 3.2 точками изображены точки задания экспериментальных данных, а сплошной кривой показан график кубического многочлена, вычисленного по МНК. Видно, что отклонения положительного и отрицательного знака более или менее уравновешены, как и следовало ожидать. Сравнение вычисленной кривой с экспериментальными значениями показывает, что наибольшее отклонение имеет место при 0^oC и равно здесь 0,0012. Все остальные отклонения существенно меньше.

Если ошибки такого порядка допустимы, то можно использовать полученную формулу в исследованиях, где требуется зависимость теплоемкости воды от температуры.

Таблица 3.1

T, ^oC	c_p	T, ^oC	c_p
0	1,00762	55	0,99919
5	1,00392	60	0,99967
10	1,00153	65	1,00024
15	1,00000	70	1,00091
20	0,99907	75	1,00167
25	0,99852	80	1,00253
30	0,99826	85	1,00351
35	0,99818	90	1,00461
40	0,99828	95	1,00586
45	0,99849	100	1,00721
50	0,99878		

§3.4. Решение несовместных систем уравнений

Методу наименьших квадратов можно дать другую интерпретацию.

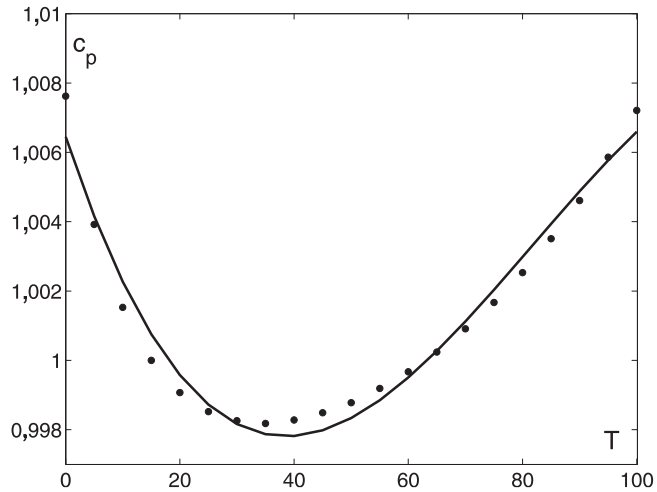


Рис. 3.2. Экспериментальные значения теплоемкости воды c_p в зависимости от температуры T (черные точки) и график кубического многочлена, построенного для этих данных по МНК.

Пусть требуется решить систему линейных алгебраических уравнений

$$\sum_{j=1}^{M} a_{ij} x_j = b_i, \quad i = 0, 1, \ldots, N, \tag{3.3}$$

где $N + 1 > M$, т. е. число уравнений больше числа неизвестных.

Взяв некоторый вектор $\mathbf{x} = (x_1, x_2, \ldots, x_M)$, образуем из уравнений (3.3) невязки

$$r_i = \sum_{j=1}^{M} a_{ij} x_j - b_i, \quad i = 0, 1, \ldots, N.$$

Если невозможно найти такой вектор $\mathbf{x}$, что все невязки будут равны нулю, то система уравнений (3.3) называется *несовместной*. Такую систему однако можно решить методом наименьших квадратов.

Будем искать минимум суммы квадратов невязок

$$E_M \equiv E(x_1, x_2, \ldots, x_M) = \sum_{i=1}^{M} r_i^2.$$

Необходимое условие экстремума $\partial E_M / \partial x_k = 0$, $k = 1, 2, \ldots, M$ дает нам систему нормальных уравнений:

$$\sum_{j=1}^{M} \Big(\sum_{i=0}^{N} a_{ij} a_{ik} \Big) x_j = \sum_{i=0}^{N} b_i a_{ik}, \quad k = 1, 2, \ldots, M. \tag{3.4}$$

Это уже система M уравнений с M неизвестными x_k. Можно показать, что система (3.4) совместна, если столбцы матрицы исходной системы (3.3) являются линейно независимыми.

Пусть $\mathbf{Ax} = \mathbf{b}$ – матричная запись исходной системы (3.3). Тогда нормальную систему (3.4) можно записать в виде:

$$\mathbf{A}^T\mathbf{Ax} = \mathbf{A}^T\mathbf{b}. \tag{3.5}$$

Систему (3.5) можно решить методом исключения Гаусса и тогда мы получим решение системы (3.3) в смысле метода наименьших квадратов.

Из (3.5) получаем $\mathbf{x} = (\mathbf{A}^T\mathbf{A})^{-1}\mathbf{A}^T\mathbf{b}$. Матрицу $\mathbf{A}^+ = (\mathbf{A}^T\mathbf{A})^{-1}\mathbf{A}^T$ называют nсевдообратной матрицей для $\mathbf{A}$. К сожалению, матрица системы (3.5) обладает плохими свойствами. Покажем это на простом примере.

Пример 3.3. Рассмотрим аппроксимацию по МНК с помощью прямой линии следующих данных.

Tаблица 3.3

x_i	15,0	15,1	15,2	15,3	15,4	15,5
y_i	33,0	33,2	33,4	33,6	33,8	34,0

Так как приведенные в табл. 3.3 данные лежат на прямой $y = 3 + 2x$, то полученная по МНК прямая $y = a_0 + a_1x$ должна иметь коэффициенты $a_0 = 3$ и $a_1 = 2$. Нам нужно решить в смысле МНК систему $\mathbf{Ax} = \mathbf{b}$, где

$$\mathbf{A} = \begin{pmatrix} 1,0 & 15,0 \\ 1,0 & 15,1 \\ 1,0 & 15,2 \\ 1,0 & 15,3 \\ 1,0 & 15,4 \\ 1,0 & 15,5 \end{pmatrix}, \quad \mathbf{b} = \begin{pmatrix} 33,0 \\ 33,2 \\ 33,4 \\ 33,6 \\ 33,8 \\ 34,0 \end{pmatrix}.$$

Нормальные уравнения МНК дают линейную систему

$$\mathbf{A}^T\mathbf{Aa} = \begin{pmatrix} 6,00000 & 91,5000 \\ 91,5000 & 1395,55 \end{pmatrix} \begin{pmatrix} a_0 \\ a_1 \end{pmatrix} = \begin{pmatrix} 201,000 \\ 3065,60 \end{pmatrix} = \mathbf{A}^T\mathbf{b}.$$

Пусть $\mathbf{B} = \mathbf{A}^T\mathbf{A}$ и $\mathbf{z} = \mathbf{A}^T\mathbf{b}$. Тогда из равенства $\mathbf{a} = \mathbf{B}^{-1}\mathbf{z}$ получаем $a_0 = 3$ и $a_1 = 2$. Заменим теперь в $\mathbf{z}$ элемент 3065,60 на 3065,59 и обозначим новую правую часть через $\tilde{\mathbf{z}}$. Из равенства $\mathbf{a} = \mathbf{B}^{-1}\tilde{\mathbf{z}}$ получаем $a_0 = 3,87$ и $a_1 = 1,94$. Таким образом, вместо прямой $y = 3 + 2x$ получаем прямую $y = 3,87 + 1,94x$.

«Малые» изменения правой части приводят к «большим» изменениям в решении нормальных уравнений.

Таким образом, нужны методы решения несовместных и переопределенных систем, свободные от такого недостатка. Здесь возможны как минимум три подхода. Первый подход основан на ортогональной факторизации матрицы $\mathbf{A}$ в системе $\mathbf{Ax} = \mathbf{b}$ и является наиболее надежным но и весьма трудоемким. Такая факторизация матрицы будет рассмотрена в гл. 5. Второй подход основан на сингулярном разложении матрицы $\mathbf{A}$, описываемом в гл. 6. Наконец, свойства матрицы нормальных уравнений можно существенно улучшить использованием ортогональных или «почти ортогональных» обобщенных многочленов (многочлены Чебышева, Лежандра и т. д.) и сплайнов.

§3.5. Нелинейные зависимости

При обработке экспериментальных данных по МНК важен правильный выбор функциональной зависимости $y = f(x)$, который должен отвечать изучаемому физическому процессу. Такой выбор может быть отличен от многочлена и обычно состоит из двух этапов. Вначале в зависимости от типа исходных данных выбирается формула приближения и осуществляется ее линеаризация, что позволяет избежать получения нелинейных нормальных уравнений. Затем по МНК определяются коэффициенты этой формулы.

Под линеаризацией обычно понимаются такие преобразования

$$\xi = \varphi(x, y), \quad \eta = \psi(x, y),$$

что эмпирическая формула приводится к виду

$$\eta = \alpha + \beta\xi,$$

где α и β – числовые коэффициенты. Вычислив α и β по МНК и выполнив обратное преобразование, находим искомую функциональную зависимость.

Пусть, например, зависимость ищется в виде показательной функции

$$y = ax^b.$$

Непосредственное использование МНК приводит к нелинейным нормальным уравнениям. Этого затруднения можно избежать, использовав вместо экспериментальных значений их логарифмы: $\log y = \log a + b \log x$. Полагая $\xi = \log x$ и $\eta = \log y$, получаем линейную зависимость:

$$\eta = \alpha + \beta\xi, \quad \alpha = \log a, \quad \beta = b.$$

Теперь найдем минимум суммы

$$E_2 \equiv E_2(a,b) = \sum(\eta_i - \alpha - \beta\xi_i)^2.$$

Дифференцируя по α и β, приходим к системе нормальных уравнений:

$$\begin{cases} (N+1)\alpha + \Big(\sum \xi_i\Big)\beta = \sum \eta_i; \\ \Big(\sum \xi_i\Big)\alpha + \Big(\sum \xi_i^2\Big)\beta = \sum \xi_i\eta_i. \end{cases}$$

Вычислив из этой линейной системы α и β, коэффициент a находим по величине логарифма.

Приведем примеры других простых линеаризующих преобразований. При $y = a + b/x$ полагаем $\xi = 1/x$, $\eta = a + b\xi$. При $y = 1/(ax+b)$ берем $\xi = x$, $\eta = 1/y$ и т. д.

§3.6. Приближение сплайнами

При большом числе исходных данных может оказаться трудно описать их одной формулой. К тому же в этом случае обусловленность нормальной системы уравнений (3.2) сильно ухудшается. Переход к системе ортогональных многочленов спасает положение лишь частично. В этом случае целесообразно применить кусочно-полиномиальные приближения. Простыми вариантами таких приближений являются кусочно-линейная функция (ломаная) и кубический сплайн.

Введем на отрезке $[a,b]$ равномерную сетку с узлами $X_j = a + (j-1)H$, $j = 1,2,\ldots,M$, $H = (b-a)/(M-1)$. Приближение по МНК будем искать в виде:

$$S(x) = \sum_{j=1}^{M} c_j B_{k,j}(x) = \sum_{j=1}^{M} c_j B_k\Big(\frac{x - X_j}{H}\Big),$$

где базисные функции (сокращенно B-сплайны) B_k при $k = 1$ и $k = 3$ задаются формулами:

$$B_1(x) = \begin{cases} 1 - |x|, & |x| \le 1; \\ 0, & 1 \le |x|, \end{cases} \qquad B_3(x) = \begin{cases} 2/3 - x^2 + |x|^3/2, & |x| \le 1; \\ (2 - |x|)^3/6, & 1 \le |x| \le 2; \\ 0, & 2 \le |x|. \end{cases}$$

Эти функции изображены на рис. 3.3. Можно показать, что функции $B_{k,j}$, $j = 1,2,\ldots,M$ (сдвиги функций B_k относительно узлов X_j) образуют на $[a,b]$ линейно независимую систему. Нормальная система (3.2) принимает здесь вид:

$$\sum_{j=1}^{M} c_j \Big(\sum_{i=0}^{N} B_{k,j}(x_i)B_{k,l}(x_i)\Big) = \sum_{i=0}^{N} y_i B_{k,l}(x_i), \quad l = 1,2,\ldots,M. \tag{3.6}$$

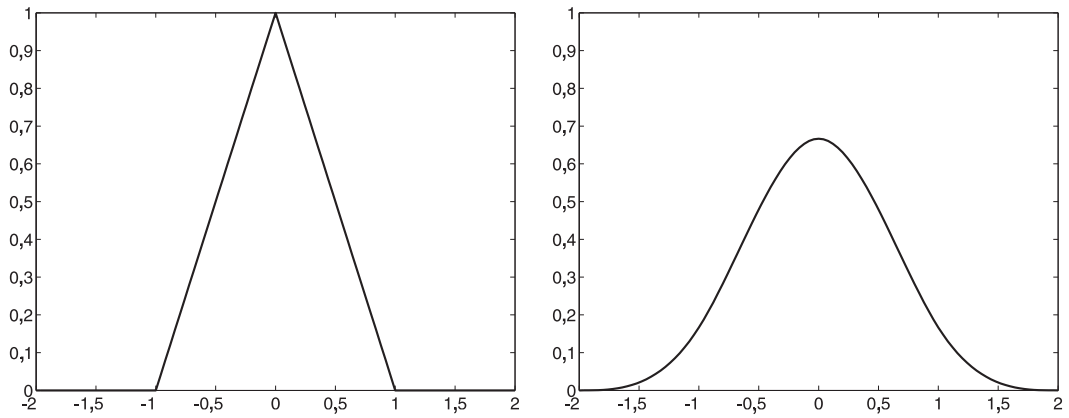


Рис. 3.3. Графики базисных сплайнов первой а) и третьей б) степени.

В силу конечности носителей B-сплайнов матрица системы (3.3) будет иметь $2k-1$ ненулевых диагоналей. Эта матрица хорошо обусловлена и положительно определена (см.[17]).

Пример 3.2. В качестве исходных данных возьмем значения функции

$$f(x) = \frac{1}{0{,}1 + (x - 0{,}2)^2} + \frac{1}{0{,}15 + (x - 0{,}8)^2}, \quad 0 \le x \le 1$$

на равномерной сетке $x_i = ih$, $i = 0, 1, \ldots, N$ с шагом $h = 1/N$, положив $N = 10$.

На рис. 3.4 а) и б) сплошной линией показан график апппроксимируемой функции f. Исходные данные помечены на нем черными кружками. Штриховой, штрих-пунктирной и пунктирной кривыми показаны графики сплайнов первой а) и третьей б) степени наилучшего среднеквадратического приближения с числом узлов $M = 2, 3, 10$. Случай $M = 2$ отвечает аппроксимации прямой линией (кубическим многочленом). При $M = 3$ ломаная (кубический сплайн) имеет два звена и т. д. Вычисления дают следующие среднеквадратические уклонения: $E_2 = 13{,}79$, $E_3 = 9{,}55$, $E_{10} = 0{,}0067$ для ломаной и $E_2 = 7{,}30$, $E_3 = 4{,}71$, $E_{10} = 0{,}0025$ для кубического сплайна. Очевидна сходимость в среднем полученных по МНК сплайнов при увеличении числа их звеньев к исходной функции f.

§3.7. Оптимизация приближения по МНК

Наличие значительной случайной погрешности в исходных данных заставляет нас отказаться от интерполирования в пользу МНК. Если в используемом приближении

$$S(x) = \sum_{j=1}^{M} c_j \varphi_j(x)$$

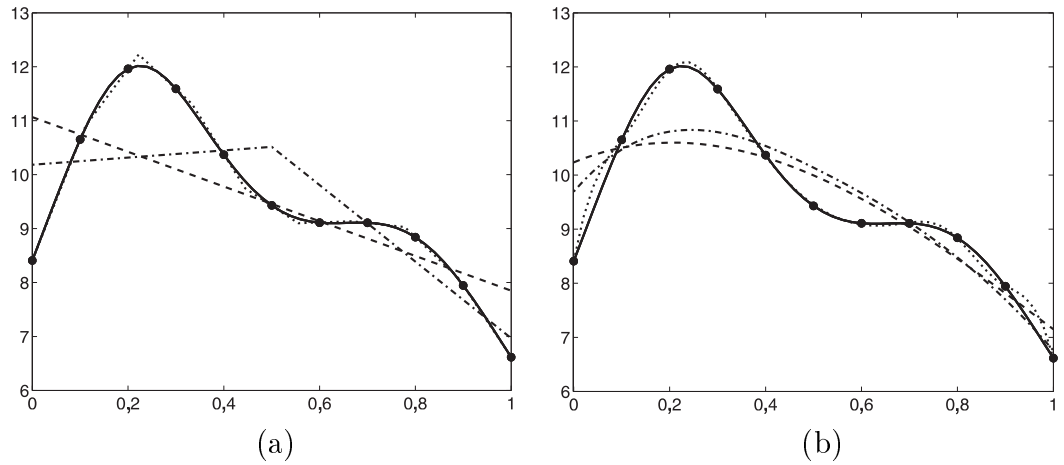


Рис. 3.4. Графики получаемых по МНК сплайнов первой а) и третьей б) степени при числе узлов: $M = 2$ (штриховая линия), $M = 3$ (штрих-пунктирная линия) и $M = 10$ (пунктирная линия). Сплошной линией показан график аппроксимируемой функции с отмеченными на нем черными точками исходными данными.

число слагаемых M мало, то следует ожидать «переглаживания», которое может существенно исказить форму исходных данных. При большом M функция S может оказаться наоборот «слишком колеблющейся».

Предположим, что известна величина погрешности исходных данных ε, т. е. они отклоняются от точных данных не более чем на эту величину. Вычислим среднюю погрешность приближения

$$\delta_M = \Big(\frac{1}{N+1}\sum_{i=0}^{N}(S(x_i) - y_i)^2\Big)^{1/2}.$$

Если $\delta_M > \varepsilon$, то погрешность аппроксимации больше погрешности входных данных и M следует увеличить. Оптимальным будет M, при котором $\delta_M \approx \varepsilon$.

Расчет можно начать с $M = 1$, когда скорее всего $\delta_M \gg \varepsilon$, и, постепенно увеличивая M, добиться выполнения условия $\delta_M \approx \varepsilon$. Если это удается сделать при $M \ll N$, то выбор базисных функций φ_j удачен. Если же $\delta_M \approx \varepsilon$ при $M \approx N$, то желательно выбрать более подходящие базисные функции φ_j. Например, вместо обычных B-сплайнов можно использовать рациональные или экспоненциальные B-сплайны или какие-либо другие функции (см. [17]).

§3.8. Изогеометрическая аппроксимация

На практике часто приходится строить аппроксимации, которые сохраняли бы геометрические свойства исходных данных такие, как положительность, монотон-

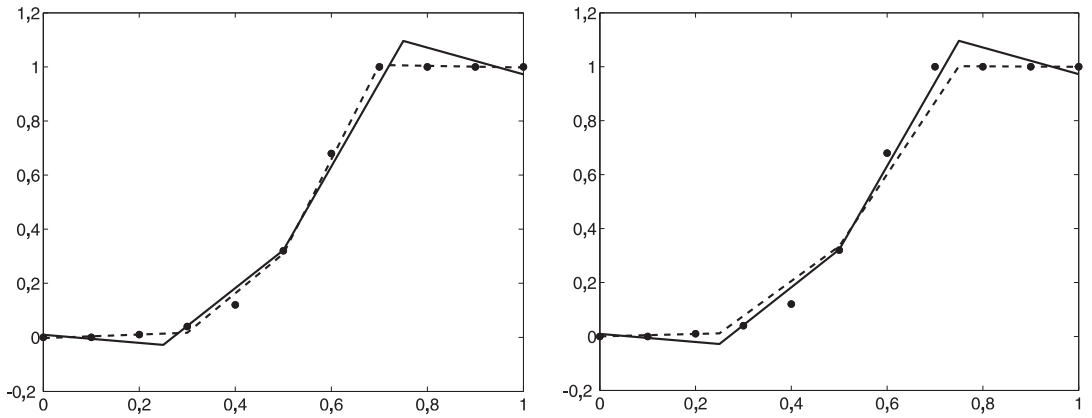


Рис. 3.5. Аппроксимация монотонно возрастающих данных ломаной по МНК. В зависимости от расположения узлов а) и выбора весовых множителей б) ломаная может не сохранять (сплошная линия) или сохранять (штриховая линия) свойство монотонности исходных данных.

ность, выпуклость, наличие прямолинейных участков и т. д. Такие приближения принято называть *изогеометрическими аппроксимациями* (см. [17]).

В табл. 3.2 приведены монотонно возрастающие данные. Аппроксимируем эти данные по МНК ломаной с равноотстоящими узлами $x_i = 0{,}25i$; $i = 0, 1, 2, 3, 4$. В результате применения МНК получаем кривую, график которой изображен на рис. 3.5 а) и б) сплошной линией. Очевидно, что эта ломаная не сохраняет свойство монотонности исходных данных. Изменим узлы ломаной на следующие: $0; 0{,}3; 0{,}5; 0{,}7; 1$. На этот раз МНК дает кривую, график которой показан на рис. 3.5 а) штриховой линией. Теперь решение по МНК монотонно возрастает в соответствии с данными. Аналогичный результат может быть получен за счет выбора в МНК весовых множителей (см. (3.1) и (3.2)). На рис. 3.5 б) штриховая линия получена при $p_0 = p_2 = p_8 = p_{10} = 100$ и $p_i = 1$ для всех остальных значений i.

Таблица 3.2

i	0	1	2	3	4	5	6	7	8	9	10
x_i	0	0,1	0,2	0,3	0,4	0,5	0,6	0,7	0,8	0,9	1
f_i	0	0	0,01	0,04	0,12	0,32	0,68	1	1	1	1

Таким образом, на форму сплайновой кривой влияет не только число звеньев сплайна, но и расположение его узлов и выбор весов. Последние можно рассматривать как дополнительные параметры для управления формой сплайна. Сплайны, узлы которых варьируются (выбираются из условий удовлетворения тем или

иным ограничениям), принято называть *сплайнами со свободными узлами*. Существуют методы оптимизации расположения узлов сплайна. Эти методы связаны однако с решением нелинейных систем уравнений и являются достаточно дорогостоящими.

§3.9. МНК и регуляризация

В основе *метода регуляризации* лежат соображения о сглаживании приближаемых данных. Распространенной формой метода регуляризации является следующая. Пусть имеются начальные данные (x_i, y_i), $i = 0, 1, \ldots, N$, где $a = x_0 < x_1 < \ldots < x_N = b$. Задавшись некоторой системой линейно независимых на отрезке $[a, b]$ функций $\{\varphi_j\}$, приближение ищем в виде:

$$S(x) = \sum_{j=0}^{N} c_j \varphi_j(x).$$

Коэффициенты c_j функции S выбираются из условия минимума выражения

$$J_\alpha(S) = \sum_{i=0}^{N} p_i \big(S(x_i) - y_i\big)^2 + \alpha I(S), \quad p_i > 0, \quad \alpha > 0, \tag{3.7}$$

где α – *параметр регуляризации* (сглаживания). Функционал I подбирается из следующего условия: если значение этого функционала невелико, то функция S обладает определенной гладкостью. Например, при

$$I(S) = \int_a^b (S')^2\, dx \quad \text{или} \quad I(S) = \int_a^b (S'')^2\, dx$$

минимизируется градиент или линеаризованная кривизна.

Пусть минимум функционала J_α в (3.7) достигается при некоторых $c_0^\alpha, c_1^\alpha, \ldots, c_N^\alpha$ и $S_\alpha(x) = \sum_{j=0}^N c_j^\alpha \varphi_j(x)$. Рассмотрим крайние случаи: $\alpha = 0$ и α – очень большое число. При $\alpha = 0$ минимум функционала (3.7) достигается на функции S_0, интерполирующей начальные данные. В этом случае

$$S_0(x_i) = \sum_{j=0}^{N} c_j^0 \varphi_j(x_i) = y_i, \quad i = 0, 1, \ldots, N$$

и в силу линейной независимости функций φ_j на $[a, b]$ решение этой системы существует и единственно.

При очень больших значениях α в функционале (3.7) определяющим является второе слагаемое, нижняя грань которого достигается на гладкой функции.

Следовательно, можно ожидать, что для промежуточных значений α решение S_α будет достаточно гладким и одновременно не будет слишком сильно уклоняться от исходных данных.

Далее в качестве практического примера регуляризации рассматривается задача построения сглаживающего кубического сплайна, реализующего компромисс между интерполяционным кубическим сплайном ($\alpha = 0$) и прямой линией ($\alpha = \infty$).

§3.10. Экстремальные свойства кубических сплайнов

Обозначим через $L_2[a,b]$ множество измеримых на отрезке $[a,b]$ функций со скалярным произведением

$$(f,g)_{L_2} = \int_a^b f(x)g(x)dx.$$

Рассмотрим класс $W_2^2[a,b]$ непрерывно дифференцируемых на $[a,b]$ функций, имеющих суммируемые с квадратом вторые производные, т. е.

$$\|f''\|_{L_2}^2 = \int_a^b [f''(x)]^2 dx < \infty. \tag{3.8}$$

Пусть в узлах сетки $\Delta : a = x_0 < x_1 < \ldots < x_N = b$ заданы некоторые числа y_i, $i = 0, 1, \ldots, N$. Поставим задачу нахождения функции $f \in W_2^2[a,b]$, удовлетворяющей условиям интерполяции

$$f(x_i) = y_i, \quad i = 0, 1, \ldots, N \tag{3.9}$$

и минимизирующей функционал (3.8).

В качестве множества допустимых функций, на котором ищется минимум функционала (3.8), могут быть взяты также подпространства $\overline{W}_2^2[a,b]$ функций, удовлетворяющих краевым условиям вида

$$f'(a) = y_0', \quad f'(b) = y_N' \tag{3.10}$$

и $\widetilde{W}_2^2[a,b]$ – периодических функций с периодом $b-a$.

Теорема 3.1. *Среди всех функций $f \in W_2^2[a,b]$, удовлетворяющих условиям интерполяции* (3.9)*, кубический сплайн S с «естественными» краевыми условиями*

$$S''(a) = S''(b) = 0 \tag{3.11}$$

является единственной функцией, минимизирующей функционал (3.8).

Если $f \in \overline{W}_2^2[a,b]$ *или* $f \in \widetilde{W}_2^2[a,b]$, *то минимум функционалу* (3.8) *доставляет сплайн из того же множества и это решение единственно.*

Доказательство. Рассмотрим скалярное произведение

$$(f'' - S'', S'')_{L_2} = [f'(x) - S'(x)]S''(x)|_a^b - (f' - S', S''')_{L_2}.$$

Первое слагаемое справа в этом равенстве будет равно нулю как в случае краевых условий (3.11), так и для функций из множеств $\overline{W}_2^2[a,b]$ и $\widetilde{W}_2^2[a,b]$. Так как кроме того $S'''(x) = c_i = const$ для $x \in (x_i, x_{i+1})$, то с учетом условий интерполяции (3.9) имеем:

$$(f'' - S'', S'')_{L_2} = -\sum_{i=0}^{N-1} c_i[f(x) - S(x)]|_{x_i}^{x_{i+1}} = 0. \tag{3.12}$$

Запишем тождество

$$\|f'' - S''\|_{L_2}^2 = \|f''\|_{L_2}^2 - 2(f'' - S'', S'')_{L_2} - \|S''\|_{L_2}^2. \tag{3.13}$$

Отсюда, учитывая равенство (3.12), получаем

$$\|S''\|_{L_2}^2 = \|f''\|_{L_2}^2 - \|f'' - S''\|_{L_2}^2 \le \|f''\|_{L_2}^2.$$

Таким образом, интерполяционный сплайн S доставляет минимум функционалу (3.8). Всякое другое решение задачи минимизации может отличаться от S лишь на многочлен первой степени, который должен удовлетворять нулевым интерполяционным условиям (3.9) и поэтому тождественно равен нулю. Теорема доказана.

Следствие 3.1. *В случае любого из множеств* $W_2^2[a,b]$, $\overline{W}_2^2[a,b]$ *или* $\widetilde{W}_2^2[a,b]$ *вторая производная интерполяционного сплайна* S *реализует наилучшее среднеквадратичное приближение второй производной интерполируемой функции* f *на множестве* $S_3(\Delta)$ *кубических сплайнов с узлами на сетке* Δ, *т. е.*

$$\|f'' - S''\|_{L_2} \le \|f'' - g''\|_{L_2} \quad \text{для всех} \quad g \in S_3(\Delta),$$

где равенство достигается в том и только том случае, если $g(x) \equiv S(x)$.

Для доказательства достаточно взять в качестве f в (3.13) разность $f - g$ и воспользоваться тем свойством, что интерполяционный сплайн от сплайна совпадает с последним.

§3.11. Минимум регуляризирующего функционала

Рассмотрим задачу нахождения функции $f \in W_2^2[a,b]$, минимизирующей функционал:

$$J_\alpha(f) = \sum_{i=0}^{N} p_i[f(x_i) - y_i]^2 + \alpha\|f''\|_{L_2}^2, \tag{3.14}$$

где p_i и α – некоторые положительные числа. Формально p_i могут принимать бесконечные значения, что дает случай интерполяции в соответствующих узлах. Выбор *параметра сглаживания* α существенно влияет на плавность получаемой кривой. *Весовые множители* p_i обычно выбирают экспериментально. Очевидно, что чем точнее измерено y_i, тем больше должно быть p_i. В этом случае функция f проходит ближе к заданному значению y_i.

Теорема 3.2. *Среди всех функций $f \in W_2^2[a,b]$ кубический сплайн с краевыми условиями* (3.10) *является единственной функцией, доставляющей минимум функционалу* (3.14).

Если $f \in \overline{W}_2^2[a,b]$ или $f \in \widetilde{W}_2^2[a,b]$, то минимум функционалу (3.14) *доставляет сплайн из того же множества и это решение единственно.*

Доказательство Пусть функция $f \in W_2^2[a,b]$ минимизирует функционал (3.14) и не является кубическим сплайном. Построим кубический сплайн S с краевыми условиями (3.11), интерполирующий f на сетке Δ. Тогда первое слагаемое в (3.14) будет одинаковым для S и f, а второе по теореме 3.1 будет меньше на S. Таким образом, $J_\alpha(S) < J_\alpha(f)$, что противоречит условию минимизации. Единственность сглаживающего сплайна также следует из теоремы 3.1. Доказательство утверждения теоремы для множеств $\overline{W}_2^2[a,b]$ и $\widetilde{W}_2^2[a,b]$ проводится аналогично. Теорема доказана.

В силу теоремы 3.2 минимум функционала (3.14) на множествах $W_2^2[a,b]$ и $\overline{W}_2^2[a,b]$ следует искать в виде кубического сплайна S_α с краевыми условиями (3.11) и (3.10) соответственно. В случае пространства $\widetilde{W}_2^2[a,b]$ сплайн S_α должен быть периодической функцией с периодом $b-a$.

§3.12. Построение сглаживающего сплайна

Рассмотрим алгоритм построения сплайна S_α, одновременно доказав его существование и единственность.

Так как для $x \in [x_i, x_{i+1}]$ вторая производная кубического сплайна S''_α – линейная функция, то при обозначении $M_j = S''_\alpha(x_j)$, $j = i, i+1$ получаем

$$S''_\alpha(x) = M_i\frac{x_{i+1}-x}{h_i} + M_{i+1}\frac{x-x_i}{h_i}, \quad i = 0,1,\ldots,N-1. \tag{3.15}$$

Находим

$$\begin{aligned}\|S''_\alpha\|^2_{L_2} &= \sum_{i=0}^{N-1}\int_{x_i}^{x_{i+1}}\Big(M_i\frac{x_{i+1}-x}{h_i}+M_{i+1}\frac{x-x_i}{h_i}\Big)^2dx = \\ &= \sum_{i=0}^{N-1}\frac{h_i}{3}(M_i^2+M_iM_{i+1}+M_{i+1}^2) = \\ &= \frac{1}{6}\sum_{i=1}^{N-1}M_i[h_{i-1}M_{i-1}+2(h_{i-1}+h_i)M_i+h_iM_{i+1}]+ \\ &\quad +\frac{h_0}{6}(2M_0+M_1)M_0+\frac{h_{N-1}}{6}(M_{N-1}+2M_N)M_N=\frac{1}{6}(\mathbf{AM},\mathbf{M}),\end{aligned}$$

где при обозначении $\alpha_i = 2(h_{i-1}+h_i)$ имеем:

$$\mathbf{A}=\begin{pmatrix}2h_0 & \gamma h_0 & 0 & \dots & 0\\ \gamma h_0 & \alpha_1 & h_1 & \dots & 0\\ \vdots & \ddots & \ddots & \ddots & \vdots\\ 0 & \dots & h_{N-2} & \alpha_{N-1} & \gamma h_{N-1}\\ 0 & \dots & 0 & \gamma h_{N-1} & 2h_{N-1}\end{pmatrix},\quad \mathbf{M}=\begin{pmatrix}M_0\\ M_1\\ \vdots\\ M_{N-1}\\ M_N\end{pmatrix}. \tag{3.16}$$

В матрице $\mathbf{A}$ коэффициент $\gamma = 0$ в случае краевых условий (3.11) и $\gamma = 1$ при краевых условиях (3.10).

Таким образом, полагая $\tilde{y}_i = S_\alpha(x_i)$, для функционала (3.14) получаем

$$J_\alpha(S_\alpha)=\sum_{i=0}^{N}p_i(\tilde{y}_i-y_i)^2+\frac{\alpha}{6}(\mathbf{AM},\mathbf{M}). \tag{3.17}$$

Для сплайна S_α с краевыми условиями (3.10) или (3.11) системы (2.38) и (2.39) из гл. 2 могут быть переписаны в виде

$$\frac{1}{6}\mathbf{AM}=\mathbf{H}\tilde{\mathbf{y}}+\bar{\mathbf{y}}', \tag{3.18}$$

где при обозначении $\beta_i = -(h_{i-1}^{-1}+h_i^{-1})$ квадратная $(N+1)\times(N+1)$ матрица $\mathbf{H}$ и векторы $\tilde{\mathbf{y}}$, $\bar{\mathbf{y}}'$ имеют вид:

$$\mathbf{H}=\begin{pmatrix}-\gamma h_0^{-1} & \gamma h_0^{-1} & 0 & \dots & 0\\ h_0^{-1} & \beta_1 & h_1^{-1} & \dots & 0\\ \vdots & \ddots & \ddots & \ddots & \vdots\\ 0 & \dots & h_{N-2}^{-1} & \beta_{N-1} & h_{N-1}^{-1}\\ 0 & \dots & 0 & \gamma h_{N-1}^{-1} & -\gamma h_{N-1}^{-1}\end{pmatrix}, \tag{3.19}$$

$$\tilde{\mathbf{y}}=(\tilde{y}_0,\tilde{y}_1,\dots,\tilde{y}_N)^T,\quad \bar{\mathbf{y}}'=(-\gamma y'_0,0,\dots,0,\gamma y'_N)^T.$$

Здесь верхний индекс T обозначает операцию транспонирования.

Согласно (3.16), матрица $\mathbf{A}$ – симметрическая с диагональным преобладанием, причем диагональные элементы положительны. Следовательно, по теореме 4.1 (Гершгорина) все ее собственные числа положительны, а сама она положительно определена. В силу (3.18) вектор M линейно выражается через вектор $\tilde{y}$. Поэтому $J_\alpha(S_\alpha)$ – положительно определенная квадратичная форма от $\tilde{\mathbf{y}}$. В качестве экстремума у нее может быть только минимум, для нахождения которого необходимо приравнять частные производные от J_α по $\tilde{y}_i$ нулю. Используя равенство (3.18), получаем

$$\begin{aligned}\frac{1}{6}\frac{\partial}{\partial\tilde{y}_i}(\mathbf{AM},\mathbf{M}) &= \frac{1}{3}\Big(\frac{\partial(\mathbf{AM})}{\partial\tilde{y}_i},\mathbf{M}\Big)=\frac{1}{3}\Big(\frac{\partial(6\mathbf{H}\tilde{\mathbf{y}}+6\bar{\mathbf{y}}')}{\partial\tilde{y}_i},\mathbf{M}\Big)= \\ &= 2\Big(\frac{\partial\tilde{\mathbf{y}}}{\partial\tilde{y}_i},\mathbf{H}^T\mathbf{M}\Big)=2(\mathbf{H}^T\mathbf{M})_i,\end{aligned}$$

где нижний индекс i обозначает i-ю компоненту вектора $\mathbf{H}^T\mathbf{M}$.

Таким образом, дифференцируя функционал в формуле (3.17), приходим к системе уравнений:

$$\frac{\partial J_\alpha}{\partial\tilde{y}_i}=2p_i(\tilde{y}_i-y_i)+2\alpha(\mathbf{H}^T\mathbf{M})_i=0,\quad i=0,1,\ldots,N. \tag{3.20}$$

После деления каждого из этих уравнений на $2p_i$ получаем

$$\alpha\mathbf{P}\mathbf{H}^T\mathbf{M}+\tilde{\mathbf{y}}=\mathbf{y}, \tag{3.21}$$

где диагональная матрица $\mathbf{P}$ и вектор $\mathbf{y}$ имеют вид:

$$\mathbf{P}=\begin{pmatrix}p_0^{-1}&0&\ldots&0\\0&p_1^{-1}&\ldots&0\\\vdots&\vdots&\ddots&\vdots\\0&0&\ldots&p_N^{-1}\end{pmatrix},\quad \mathbf{y}=\begin{pmatrix}y_0\\y_1\\\vdots\\y_N\end{pmatrix}.$$

Покажем, что соотношения (3.21) с краевыми условиями (3.11) или (3.10) являются достаточными условиями минимума функционала J_α на множествах $W_2^2[a,b]$ и $\bar{W}_2^2[a,b]$ соответственно.

Так как, согласно (3.15), имеем:

$$S_\alpha'''(x)=\frac{M_{i+1}-M_i}{h_i},\quad x\in(x_i,x_{i+1}),\quad i=0,1,\ldots,N-1,$$

то условия минимума (3.20) можно переписать в виде:

$$\alpha D_i+p_i(\tilde{y}_i-y_i)=0,\quad i=0,1,\ldots,N, \tag{3.22}$$

где

$$D_i = \begin{cases} S'''(x_0+0), & i=0; \\ S'''(x_i+0) - S'''(x_i-0), & i=1,2,\ldots,N-1; \\ -S'''(x_N-0) & i=N. \end{cases}$$

Рассмотрим функционал

$$\begin{aligned} \tilde{J}_\alpha(f-S_\alpha) &= \sum_{i=0}^{N} p_i[f(x_i)-S_\alpha(x_i)]^2 + \alpha\|f''-S''_\alpha\|^2_{L_2} = \\ &= J_\alpha(f) - J_\alpha(S_\alpha) - 2I, \end{aligned}$$

где

$$I = \alpha(f''-S''_\alpha, S''_\alpha)_{L_2} - \sum_{i=0}^{N} p_i[y_i - S_\alpha(x_i)][f(x_i)-S_\alpha(x_i)].$$

Учитывая равенства (3.22) и краевые условия (3.11) или (3.10), получаем:

$$\begin{aligned} I &= \alpha\Big\{(f''-S''_\alpha, S''_\alpha)_{L_2} - \sum_{i=0}^{N} D_i[f(x_i)-S_\alpha(x_i)]\Big\} = \\ &= \alpha\Big\{(f''-S''_\alpha, S''_\alpha)_{L_2} + \sum_{i=0}^{N-1} S'''_\alpha(x_i+0)[f(x)-S_\alpha(x)]\Big|_{x_i}^{x_{i+1}}\Big\} = \\ &= \alpha[f'(x)-S'_\alpha(x)]S''_\alpha(x)\Big|_a^b = 0. \end{aligned}$$

Таким образом,

$$\tilde{J}_\alpha(f-S_\alpha) + J_\alpha(S_\alpha) = J_\alpha(f).$$

Кроме того, поскольку $\tilde{J}_\alpha(f-S_\alpha) \geq 0$, то $J_\alpha(S_\alpha) \leq J_\alpha(f)$. Следовательно, сплайн S_α, удовлетворяющий соотношениям (3.21) и краевым условиям (3.11) или (3.10), доставляет минимум функционалу (3.14). Аналогично устанавливается, что условия (3.21) являются достаточными условиями минимума функционала J_α также в случае пространства периодических функций $\widetilde{W}_2^2[a,b]$.

Умножая равенство (3.21) слева на матрицу $\mathbf{H}$ из (3.19) и учитывая соотношение (3.18), получаем систему линейных уравнений с пятидиагональной матрицей:

$$\Big(\frac{1}{6}\mathbf{A} + \alpha\mathbf{H}\mathbf{P}\mathbf{H}^T\Big)\mathbf{M} = \mathbf{H}\mathbf{y} + \bar{\mathbf{y}}'. \tag{3.23}$$

Здесь матрица $\mathbf{A}$ – симметрична и положительно определена. Пусть $\mathbf{r} = \mathbf{H}^T\tilde{\mathbf{y}}$. Так как матрица $\mathbf{P}$ – положительно полуопределена, то $\tilde{\mathbf{y}}^T(\mathbf{H}\mathbf{P}\mathbf{H}^T)\tilde{\mathbf{y}} = \mathbf{r}^T\mathbf{P}\mathbf{r} \geq 0$. Следовательно, матрица $\mathbf{H}\mathbf{P}\mathbf{H}^T = (\mathbf{H}\mathbf{P}\mathbf{H}^T)^T$ симметрична и положительно полуопределена. Матрица системы (3.23) будет положительно определена как сумма

положительно и неотрицательно определенных матриц. Поэтому она невырождена. Это доказывает существование и единственность сглаживающего кубического сплайна в классах $W_2^2[a,b]$ и $\overline{W}_2^2[a,b]$. Доказательство существования и единственности сглаживающего сплайна в подпространстве $\widetilde{W}_2^2[a,b]$ проводится аналогично.

Систему (3.23) можно решить *методом пятиточечной прогонки*. Эффективны также методы, основанные на разложении симметрической матрицы этой системы в виде $\mathbf{LDL}^T$, где $\mathbf{L}$ – нижняя треугольная матрица с единичной диагональю, а $\mathbf{D}$ – диагональная матрица с положительными элементами.

Выпишем в явном виде системы линейных уравнений, которые требуется решать при построении сглаживающего сплайна. В случае краевых условий (3.10) и (3.11) имеем:

$$\begin{aligned}
a_0M_0 + b_0M_1 + c_0M_2 &= d_0,\\
b_0M_0 + a_1M_1 + b_1M_2 + c_1M_3 &= d_1,\\
c_{i-2}M_{i-2} + b_{i-1}M_{i-1} + a_iM_i + b_iM_{i+1} + c_iM_{i+2} &= d_i,\\
i = 2,3,\ldots,N-2,&\\
c_{N-3}M_{N-3} + b_{N-2}M_{N-2} + a_{N-1}M_{N-1} + b_{N-1}M_N &= d_{N-1},\\
c_{N-2}M_{N-2} + b_{N-1}M_{N-1} + a_NM_N &= d_N,
\end{aligned} \tag{3.24}$$

где коэффициенты вычисляются по формулам:

$$\begin{aligned}
a_0 &= \frac{h_0}{3} + \frac{\alpha\gamma}{h_0^2}(p_0^{-1} + p_1^{-1}), \quad c_0 = \frac{\alpha\gamma}{h_0h_1}p_1^{-1},\\
b_0 &= \gamma\frac{h_0}{6} - \frac{\alpha\gamma}{h_0}\Big[\frac{1}{h_0}p_0^{-1} + \Big(\frac{1}{h_0} + \frac{1}{h_1}\Big)p_1^{-1}\Big],\ d_0 = \gamma(y[x_0,x_1] - y_0'),
\end{aligned} \tag{3.25a}$$

$$\begin{aligned}
a_i &= \frac{1}{3}(h_{i-1} + h_i) + \alpha\Big[\frac{1}{h_{i-1}^2}p_{i-1}^{-1} + \Big(\frac{1}{h_{i-1}} + \frac{1}{h_i}\Big)^2 p_i^{-1} + \frac{1}{h_i^2}p_{i+1}^{-1}\Big],\\
&\qquad i = 1,2,\ldots,N-1,\\
b_i &= \frac{h_i}{6} - \frac{\alpha}{h_i}\Big[\Big(\frac{1}{h_{i-1}} + \frac{1}{h_i}\Big)p_i^{-1} + \Big(\frac{1}{h_i} + \frac{1}{h_{i+1}}\Big)p_{i+1}^{-1}\Big], \quad i = 1,2,\ldots,N-2,\\
c_i &= \frac{\alpha}{h_ih_{i+1}}p_{i+1}^{-1}, \quad i = 1,2,\ldots,N-3,\\
d_i &= y[x_i,x_{i+1}] - y[x_{i-1},x_i], \quad i = 1,2,\ldots,N-1.
\end{aligned} \tag{3.25b}$$

$$\begin{aligned}
a_N &= \frac{h_{N-1}}{3} + \frac{\alpha\gamma}{h_{N-1}^2}(p_{N-1}^{-1} + p_N^{-1}),\\
b_{N-1} &= \gamma\frac{h_{N-1}}{6} - \frac{\alpha\gamma}{h_{N-1}}\Big[\Big(\frac{1}{h_{N-2}} + \frac{1}{h_{N-1}}\Big)p_{N-1}^{-1} + \frac{1}{h_{N-1}}p_N^{-1}\Big],\\
c_{N-2} &= \frac{\alpha\gamma}{h_{N-2}h_{N-1}}p_{N-1}^{-1}, \quad d_N = \gamma(y_N' - y[x_{N-1},x_N]).
\end{aligned} \tag{3.25c}$$

В формулах (3.25a) и (3.25c) коэффициент $\gamma = 0$ для краевых условий (3.11) и $\gamma = 1$ при краевых условиях (3.10).

В периодическом случае система состоит из уравнений

$$c_{i-2}M_{i-2} \; + \; b_{i-1}M_{i-1} + a_iM_i + b_iM_{i+1} + c_iM_{i+2} = d_i; \quad i = 1, 2, \ldots, N, \tag{3.26}$$

где для всех i коэффициенты определяются формулами (3.25b). Здесь величины с индексами i и $N + i$ считаются равными, т. е. $M_i = M_{N+i}$, $h_i = h_{N+i}$, $a_i = a_{N+i}$ и т. д.

После того как найден вектор $\mathbf{M}$, вектор сеточных значений сглаживающего сплайна $\tilde{\mathbf{y}}$ находится из формуле (3.21):

$$\tilde{\mathbf{y}} = \mathbf{y} - \alpha \mathbf{P}\mathbf{H}^T\mathbf{M}$$

или, согласно (3.22), в покомпонентной форме

$$\tilde{y}_i = y_i - \frac{\alpha}{p_i}D_i, \quad i = 0, 1, \ldots, N, \tag{3.27}$$

где

$$D_0 = \frac{1}{h_0}(M_1 - M_0); \tag{3.28a}$$

$$D_i = \frac{1}{h_i}(M_{i+1} - M_i) - \frac{1}{h_{i-1}}(M_i - M_{i-1}), \quad i = 1, 2, \ldots, N-1; \tag{3.28b}$$

$$D_N = -\frac{1}{h_{N-1}}(M_N - M_{N-1}). \tag{3.28c}$$

В периодическом случае $M_N = M_0$, $M_{N+1} = M_1$, $h_N = h_0$ и все величины D_i находятся по формуле (3.28b), где $i = 1, 2, \ldots, N$.

При вычислении значений сплайна по формулам (2.36) или (2.49) из гл. 2 необходимо заменить в них f_j на $\tilde{y}_j$, $j = i, i + 1$.

§3.13. Метод пятиточечной прогонки

Построение сглаживающего сплайна требует решения системы линейных алгебраических уравнений (3.24) или (3.26) с симметрической и положительно определенной матрицей. Известно [17], что решение такой системы может быть осуществлено методом исключения Гаусса без выбора главных элементов, т. е. в нашем случае методом обычной или периодической пятиточечной прогонки [13, 24].

Имея в виду системы уравнений, возникающие при построении сглаживающего сплайна, рассмотрим следующую линейную систему

$$\begin{pmatrix} a_1 & b_1 & c_1 & 0 & \dots & c_{n-1} & b_n \\ b_1 & a_2 & b_2 & c_2 & \dots & 0 & c_n \\ c_1 & b_2 & a_3 & b_3 & \dots & 0 & 0 \\ \vdots & \ddots & \ddots & \ddots & \ddots & \vdots & \\ 0 & 0 & \dots & b_{n-3} & a_{n-2} & b_{n-2} & c_{n-2} \\ c_{n-1} & 0 & \dots & c_{n-3} & b_{n-2} & a_{n-1} & b_{n-1} \\ b_n & c_n & \dots & 0 & c_{n-2} & b_{n-1} & a_n \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \\ x_3 \\ \vdots \\ x_{n-2} \\ x_{n-1} \\ x_n \end{pmatrix} = \begin{pmatrix} d_1 \\ d_2 \\ d_3 \\ \vdots \\ d_{n-2} \\ d_{n-1} \\ d_n \end{pmatrix}. \tag{3.29}$$

Предположим вначале, что $c_{n-1} = b_n = c_n = 0$. Это отвечает случаю системы с пятидиагональной матрицей (3.24). Чтобы начать исключение, разделим первое уравнение этой системы на диагональный элемент a_1 и обозначим

$$p_1 = b_1/a_1, \quad q_1 = c_1/a_1, \quad r_1 = d_1/a_1. \tag{3.30}$$

Умножая первое уравнение системы на b_1 и вычитая его из второго, исключим поддиагональный элемент b_1 во второй строке. Разделив затем второе уравнение на диагональный элемент $a_2 - p_1 b_1$, обозначим

$$p_2 = \frac{b_2 - q_1 b_1}{a_2 - p_1 b_1}, \quad q_2 = \frac{c_2}{a_2 - p_1 b_1}, \quad r_2 = \frac{d_2 - r_1 b_1}{a_2 - p_1 b_1}. \tag{3.31}$$

Предположим, что мы исключили все ненулевые поддиагональные элементы в первых $i-1$ строках. Подставляя теперь неизвестные

$$x_j = r_j - p_j x_{j+1} - q_j x_{j+2}, \quad j = i-2, i-1$$

в i-ю строку системы, имеем:

$$\begin{aligned} & c_{i-2}x_{i-2} + b_{i-1}x_{i-1} + a_i x_i + b_i x_{i+1} + c_i x_{i+2} = \\ = \; & c_{i-2}[r_{i-2} - p_{i-2}(r_{i-1} - p_{i-1}x_i - q_{i-1}x_{i+1}) - q_{i-2}x_i] \; + \\ & + b_{i-1}(r_{i-1} - p_{i-1}x_i - q_{i-1}x_{i+1}) + a_i x_i + b_i x_{i+1} + c_i x_{i+2} = \\ = \; & [a_i - p_{i-1}(b_{i-1} - p_{i-2}c_{i-2}) - q_{i-2}c_{i-2}]x_i \; + \\ & + [b_i - q_{i-1}(b_{i-1} - p_{i-2}c_{i-2})]x_{i+1} + c_i x_{i+2} \; + \\ & + r_{i-1}(b_{i-1} - p_{i-2}c_{i-2}) + r_{i-2}c_{i-2} = d_i. \end{aligned}$$

Отсюда следуют рекуррентные формулы:

$$p_i \;\; = \;\; \frac{b_i - q_{i-1}\beta_i}{\alpha_i}, \quad i = 3, 4, \dots, n-1;$$

$$
\begin{aligned}
q_i &= \frac{c_i}{\alpha_i}, \quad i = 3, 4, \ldots, n-2; \\
r_i &= \frac{d_i - r_{i-1}\beta_i - r_{i-2}c_{i-2}}{\alpha_i}, \quad i = 3, 4, \ldots, n,
\end{aligned} \tag{3.32}
$$

где

$$
\alpha_i = a_i - p_{i-1}\beta_i - q_{i-2}c_{i-2}; \quad \beta_i = b_{i-1} - p_{i-2}c_{i-2}; \quad i = 3, 4, \ldots, n.
$$

Формулы (3.30)–(3.32) позволяют легко вычислить коэффициенты p_i, q_i и r_i и тем самым привести систему (3.29) к верхнему треугольному виду

$$
\begin{pmatrix}
1 & p_1 & q_1 & 0 & \ldots & 0 \\
0 & 1 & p_2 & q_2 & \ldots & 0 \\
\vdots & \ddots & \ddots & \ddots & \ddots & \vdots \\
0 & \ldots & 0 & 1 & p_{n-2} & q_{n-2} \\
0 & \ldots & 0 & 0 & 1 & p_{n-1} \\
0 & \ldots & 0 & 0 & 0 & 1
\end{pmatrix}
\begin{pmatrix} x_1 \\ x_2 \\ \vdots \\ x_{n-2} \\ x_{n-1} \\ x_n \end{pmatrix}
=
\begin{pmatrix} r_1 \\ r_2 \\ \vdots \\ r_{n-2} \\ r_{n-1} \\ r_n \end{pmatrix}.
$$

После этого, используя обратный ход, находим решение системы (3.29):

$$
\begin{aligned}
x_n &= r_n, \quad x_{n-1} = r_{n-1} - p_{n-1}r_n, \\
x_i &= r_i - p_i x_{i+1} - q_i x_{i+2}, \quad i = n-2, n-3, \ldots, 1.
\end{aligned} \tag{3.33}
$$

§3.14. Корректность и устойчивость пятиточечной прогонки

Пусть для элементов матрицы системы (3.29) выполняются условия строгого диагонального преобладания

$$
|a_i| > |c_{i-2}| + |b_{i-1}| + |b_i| + |c_i|, \quad i = 1, \ldots, n, \tag{3.34}
$$

где $b_0 = c_{-1} = c_0 = b_n = c_{n-1} = c_n = 0$.

Покажем, что при выполнении неравенств (3.34) алгоритм пятиточечной прогонки корректен и устойчив.

Согласно формулам (3.30) и (3.31) и неравенствам (3.34), имеем

$$
\begin{aligned}
|p_1| + |q_1| &= \frac{|b_1| + |c_1|}{|a_1|} < 1, \\
|p_2| + |q_2| &= \frac{|b_2 - q_1 b_1| + |c_2|}{|a_2 - p_1 b_1|} < \frac{|b_2| + (1 - |p_1|)|b_1| + |c_2|}{|a_2| - |p_1||b_1|} < 1.
\end{aligned}
$$

Предположим по индукции, что $|p_j| + |q_j| < 1$, $j = 1, 2, \ldots, i-1$. Тогда, используя формулы (3.32) и условия диагонального преобладания (3.34), получаем:

$$\begin{aligned} |p_i| + |q_i| &= \frac{|b_i - q_{i-1}\beta_i| + |c_i|}{|a_i - p_{i-1}\beta_i - q_{i-2}c_{i-2}|} < \frac{|b_i| + (1 - |p_{i-1}|)|\beta_i| + |c_i|}{|a_i| - |p_{i-1}||\beta_i| - |q_{i-2}||c_{i-2}|} < \\ &< \frac{|b_i| + |b_{i-1}| + (1 - |q_{i-2}|)|c_{i-2}| + |c_i| - |p_{i-1}||\beta_i|}{|a_i| - |p_{i-1}||\beta_i| - |q_{i-2}||c_{i-2}|} < 1. \end{aligned}$$

Таким образом, $|p_i| + |q_i| < 1$ для $i = 1, \ldots, n-2$ и $|p_{n-1}| < 1$.

Так как, согласно неравенствам (3.34), имеем

$$\begin{aligned} |a_1| &> 0, \quad |a_2 - p_1 b_1| \geq |a_2| - |p_1||b_1| > |a_2| - |b_1| > 0, \\ |\alpha_i| &= |a_i - p_{i-1}(b_{i-1} - p_{i-2}c_{i-2}) - q_{i-2}c_{i-2}| \geq \\ &\geq |a_i| - |c_{i-2}|(|q_{i-2}| + |p_{i-1}||p_{i-2}|) - |p_{i-1}||b_{i-1}| > \\ &> |a_i| - |c_{i-2}| - |b_{i-1}| > 0, \quad i = 3, 4, \ldots, n, \end{aligned} \tag{3.35}$$

знаменатели в формулах (3.30)–(3.32) отличны от нуля, что означает выполнимость всех используемых при реализации прогонки формул, т. е. корректность метода пятиточечной прогонки.

Пусть вычисления дают приближенное решение $\bar{x}_i = x_i + \varepsilon_i$, $i = n-2, \ldots, 1$, где ε_i – ошибка округления на i-м шаге. Тогда, согласно формулам (3.33), имеем:

$$\bar{x}_i = r_i - p_i\bar{x}_{i+1} - q_i\bar{x}_{i+2}; \quad i = n-2, n-3, \ldots, 1.$$

Вычитая из этого уравнения соотношение (3.33), находим

$$\varepsilon_i = -p_i\varepsilon_{i+1} - q_i\varepsilon_{i+2}; \quad i = n-2, n-3, \ldots, 1,$$

откуда

$$|\varepsilon_i| = |p_i||\varepsilon_{i+1}| + |q_i||\varepsilon_{i+2}| < \max(|\varepsilon_{i+1}|, |\varepsilon_{i+2}|); \quad i = n-2, n-3, \ldots, 1,$$

т. е. алгоритм пятиточечной прогонки является устойчивым.

Алгоритм периодической пятиточечной прогонки, его корректность и устойчивость подробно рассмотрены в [17].

§3.15. Выбор весовых множителей

Величина локального уклонения сглаживающего сплайна от заданных значений регулируется с помощью задания весовых множителей p_i в функционале (3.14). Как уже отмечалось в § 3.2, чем точнее измерено y_i, тем больше должно

быть значение веса p_i. В частности, при $p_i = \infty$ значение y_i будет закреплено, т. е. оно будет интерполироваться.

Предположим, что исходные данные находятся в «коридоре»:

$$\varepsilon_i = |y_i - y_i^*| \le \delta_i; \quad i = 0, 1, \ldots, N,$$

где y_i^* – точное значение измеряемой величины. В этом случае, исходя из формул (3.27) и (3.28a)–(3.28c), естественно потребовать, чтобы

$$\frac{\alpha}{p_i}|D_i| = |\tilde{y}_i - y_i| \le \delta_i; \quad i = 0, 1, \ldots, N. \tag{3.36}$$

Следуя [13], рассмотрим следующий итерационный алгоритм нахождения весов p_i в функционале (3.14), который должен обеспечить выполнение ограничений (3.36):

$$\Big(\frac{1}{6}\mathbf{A} + \alpha\mathbf{H}\mathbf{P}^{(k)}\mathbf{H}^T\Big)\mathbf{M}^{(k)} = \mathbf{H}\mathbf{y} + \bar{\mathbf{y}}', \tag{3.37}$$

$$p_i^{(k+1)} = \frac{\alpha|D_i^{(k)}|}{\delta_i}; \quad k = 0, 1, \ldots, \tag{3.38}$$

где k – номер итерации.

Пусть теперь на k-й итерации в точке x_i условие (3.36) не выполняется

$$\varepsilon_i^{(k)} = \frac{\alpha}{p_i^{(k)}}|D_i^{(k)}| > \delta_i.$$

Тогда из формулы (3.38) следует, что

$$p_i^{(k)} < \frac{\alpha}{\delta_i}|D_i^{(k)}| = p_i^{(k+1)},$$

т. е. на $(k+1)$-й итерации весовой множитель p_i возрастает. Это ведет к уменьшению ε_i и возврату значения сплайна в «коридор».

Если на k-й итерации $\varepsilon_i^{(k)} < \delta_i$, то по формуле (3.38) имеем $p_i^{(k+1)} < p_i^{(k)}$ и значение сплайна начинает удаляться от исходной величины y_i. Это ведет к большей плавности графика сглаживающего сплайна.

Наконец, если в процессе итераций $p_i^{(k+1)} = 0$, то скачок третьей производной сплайна в точке x_i отсутствует, т. е. стыкующиеся в этой точке соседние кубические многочлены совпадают и могут быть заменены одним многочленом. Это позволяет «сжать» исходную информацию, удалив из рассмотрения лишние узлы сплайна.

В качестве начального приближения для итераций (3.37)–(3.38) берется интерполяционный сплайн и полагается $D_i^{(0)} = D_i$. Первые несколько итераций следует

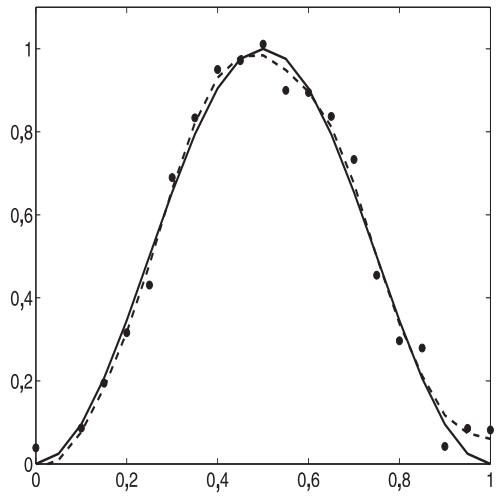


Рис. 3.6. Сглаживание кубическим сплайном с краевыми условиями $S'(0) = f'(0)$ и $S'(1) = f'(1)$ «экспериментальных» данных (черные точки). Графики исходной функции f и сглаживающего сплайна S показаны сплошной и штриховой линиями соответственно. Параметр сглаживания $\alpha = 0{,}0001$.

выполнить без учета условия (3.36). Итерационный процесс должен продолжаться до тех пор, пока значения сплайна $\tilde{y}_i$ в узлах сетки не окажутся в «коридоре» $|\tilde{y}_i - y_i| \le \delta_i$.

В процессе итераций краевые условия (3.10) могут войти в противоречие с характером поведения сплайновой кривой. Чтобы избежать этой ситуации, целесообразно значения производной сплайна $S'(a)$ и $S'(b)$ для $(k+1)$-й итерации задавать с помощью квадратических многочленов Лагранжа, интерполирующих полученные на k-й итерации значения сплайна в точках x_0, x_1, x_2 и x_{N-2}, x_{N-1}, x_N соответственно.

На рис. 3.6 приведен пример сглаживания «экспериментальных» данных, полученных 10 % зашумлением с помощью датчика случайных чисел значений функции «шапочки» $f(x) = \sin^2(\pi x)$, $0 \le x \le 1$ на равномерной сетке с шагом $h = 0{,}05$. Вначале по исходным данным, отмеченным черными кружками, строился интерполяционный кубический сплайн S с краевыми условиями $S'(0) = f'(0)$ и $S'(1) = f'(1)$. Графики исходной функции f и сглаживающего сплайна S показаны сплошной и штриховой линиями соответственно. Значения весовых множителей $p_i^{(k+1)}$, полученные по формуле (3.38), были в пределах $10^4 - 10^5$ на первой итерации и затем убывали на порядок на последующих итерациях. Использовалось значение параметра сглаживания $\alpha = 0{,}0001$.

§3.16. Выбор параметра сглаживания

При построении сглаживающего сплайна S_α возникает проблема разумного выбора параметра сглаживания α. Для выбора этого параметра обычно используется так называемый «критерий невязки» [26], когда параметр α выбирается из условия

$$\varphi(\alpha) \equiv \Big(\sum_{i=0}^{N} p_i [S_\alpha(x_i) - y_i]^2 \Big)^{1/2} = \varepsilon. \tag{3.39}$$

Здесь $\varepsilon > 0$ – допустимый «уровень» уклонения от заданных значений y_i значений сглаживающего сплайна $\tilde{y}_i = S_\alpha(x_i)$. Уравнение (3.39) существенно нелинейное. Его решение может быть найдено итерационным методом Ньютона. Сходимость итерационной последовательности $\{\alpha_k\}$ гарантируется, если начальное приближение α_0 выбрано из условия $\varphi(\alpha_0) > \varepsilon$. Если $\varphi(\alpha_k) < \varepsilon$, то α_{k+1} может стать бесконечным или отрицательным [17].

Построение сглаживающего сплайна по сравнению с интерполяционным сплайном требует значительно большего объема вычислений.

§3.17. Задачи

3.1. Пусть имеются исходные данные (x_i, y_i), $i = 0, 1, \ldots, N$. Покажите, что если с помощью МНК искать уравнение кривой в виде $y = a$, то величина a будет равна среднему арифметическому из всех значений y_i.

3.2. Покажите, что если с помощью МНК искать уравнение кривой в виде $y = a + bx$, причем исходная информация ограничена двумя точками (x_1, y_1) и (x_2, y_2), то решение системы нормальных уравнений приводит к уравнению прямой, проходящей через эти две точки.

3.3. Известно, что вязкость жидкости V изменяется с температурой T по правилу $V = a + bT + cT^2$. Найдите наилучшие значения a, b и c для следующих данных:

T	1	2	3	4	5	6	7
V	2,31	2,01	1,80	1,66	1,55	1,47	1,41

3.4. Автомобильный дилер хочет знать спрос на автомобили в зависимости от их цены. Следующая таблица показывает продажи за квартал автомобилей по четырем разным ценам.

Цена в 1000 долларов (x_i)	6,5	8,3	11,2	14,3
Спрос в количестве авто (y_i)	33	21	13	9

а) Найдите по методу наименьших квадратов прямую, приближенно описывающую данные в приведенной таблице.

б) Оцените спрос при цене 10 000 долларов.

3.5. Покажите, что с помощью соответствующего преобразования можно получить линейные нормальные уравнения для случая функциональной зависимости вида:

$$\text{а) } y = \frac{x}{ax+b}; \qquad \text{б) } y = a + \frac{b}{x} + \frac{c}{x^2}.$$

3.6. Постройте ломаную с узлами $x_i = ih$, $i = 0, 1, \ldots, N$, $h = 10/N$, приближающую по МНК приводимые ниже монотонно возрастающие данные. Положите $N = 4$. Как надо расположить узлы ломаной, чтобы она сохраняла свойство монотонности исходных данных?

i	0	1	2	3	4	5	6	7	8	9	10
x_i	0	2	3	5	6	8	9	11	12	14	15
f_i	10	10	10	10	10	10	10,5	15	56	60	85

3.7. Возьмите выборку значений функции $f(x) = e^{-x^2}$ на равномерной сетке отрезка $[-1, 1]$ с шагом $h = 2/N$ и зашумите их с помощью датчика случайных чисел на величину $\varepsilon = 0,1$. Используя кубические B-сплайны, найдите оптимальное МНК-приближение по числу привлекаемых базисных функций.

3.8. Дана несовместная система линейных уравнений

$$\begin{cases} 2x + 3y = 1; \\ x - 4y = -9; \\ 2x - y = -1. \end{cases}$$

Найдите наилучшее в смысле МНК приближенное решение.

3.9. Решите задачу 3.7, используя сглаживающий кубический сплайн.

Глава 4
Численное дифференцирование и интегрирование

Методы численного дифференцирования и интегрирования используются во многих приложениях. Эти методы важны не только сами по себе, но и как вспомогательные при построении методов численного решения дифференциальных и интегральных уравнений. Наличие большого количества таких методов объясняется их важностью в приложениях. Здесь будут рассмотрены только некоторые наиболее употребительные из этих методов, позволяющие понять основные принципы их построения.

§ 4.1. Задача численного дифференцирования

Задача численного дифференцирования возникает, когда требуется найти производную таблично заданной функции f, или аналитическое выражение для функции f является громоздким и трудновычислимым.

Покажем вначале, что задача дифференцирования в пространстве непрерывно дифференцируемых функций C^1 является некорректной. Напомним, что задача называется корректной, если ее решение существует, единственно и непрерывно зависит от входных данных. Используем следующий пример из [3]. Пусть

$$f(x) \in C^1 \quad \text{и} \quad \tilde{f}(x) = f(x) + \frac{1}{n}\sin(n^2 x),$$

где n – натуральное число. Тогда

$$\rho = \max|f(x) - \tilde{f}(x)| = \frac{1}{n}\max|\sin(n^2 x)| = \frac{1}{n},$$

т. е. $\rho \to 0$ при $n \to \infty$. В то же время

$$\rho_1 = \max|f'(x) - \tilde{f}'(x)| = n\max|\cos(n^2 x)| = n$$

и $\rho_1 \to \infty$ при $n \to \infty$.

Таким образом, нет непрерывной зависимости от начальных данных и, следовательно, задача дифференцирования в пространстве C^1 является некорректно поставленной. Как следствие некорректной является и задача численного дифференцирования. К этому следует добавить, что обычно значения таблично заданной функции являются неточными, что весьма существенно сказывается на результате решения некорректно поставленной задачи и усложняет ее решение.

§ 4.2. Методы численного дифференцирования

Численное дифференцирование таблично заданной функции может быть осуществлено различными способами. Прежде всего по табличным данным может быть построен интерполяционный/сглаживающий многочлен, сплайн и т. д., дифференцируя который можно найти нужные значения производных в заданных точках. Этот подход подробно рассматривался в гл. 2 и 3 и поэтому здесь мы не будем на нем останавливаться.

Другой способ построения формул численного дифференцирования, приводящий к тем же самым формулам, – это метод неопределенных коэффициентов. В этом случае, полагая

$$f^{(k)}(x_p) \approx \sum_{i=1}^{n} c_i f(x_i), \tag{4.1}$$

коэффициенты c_i находим из условия точности этой формулы на многочленах максимально высокой степени. Выбирая в качестве функции f мономы x^j, $j = 0, 1, \ldots, m$ и подставляя их в равенство (4.1), получаем систему линейных алгебраических уравнений:

$$j(j-1)\ldots(j-k+1)x_p^{j-k} = \sum_{i=1}^{n} c_i x_i^j; \quad j = 0, 1, \ldots, m \tag{4.2}$$

относительно неизвестных коэффициентов c_i. При $m = n - 1$ имеем систему n уравнений с n неизвестными, определитель которой является определителем Вандермонда и отличен от нуля. Следовательно, всегда можно построить формулу численного дифференцирования, точную на многочленах степени $n-1$. Для формул вида (4.1), симметричных относительно точки x_p, это удается сделать и при $m = n$, если k четное, n нечетное или k нечетное, n четное.

Следующая задача – определение погрешности формул численного дифференцирования. Это можно сделать, например, разложением в формуле (4.1) значений $f(x_i)$ в точке x_p по формуле Тейлора.

Простейшая формула численного дифференцирования

$$f'(x_i) \approx \frac{f(x_i + h) - f(x_i)}{h}$$

точна на многочленах первой степени. Так как согласно разложению по формуле Тейлора $f(x_i + h) = f(x_i) + hf'(x_i) + O(h^2)$, то

$$f'(x_i) = \frac{f(x_i + h) - f(x_i)}{h} + O(h).$$

Пусть

$$f'(x_i) \approx c_1 f(x_i - h) + c_2 f(x_i) + c_3 f(x_i + h).$$

Система (4.2) здесь принимает вид:

$$\begin{cases} c_1 + c_2 + c_3 = 0; \\ -hc_1 + hc_3 = 1; \\ h^2 c_1 + h^2 c_3 = 0. \end{cases}$$

откуда $c_1 = -c_3 = 1/(2h)$; $c_2 = 0$, т. е.

$$f'(x_i) \approx \frac{f(x_i + h) - f(x_i - h)}{2h}.$$

Пользуясь теперь разложением по формуле Тейлора значений $f(x_i \pm h)$ в точке x_i, получаем

$$f(x_i \pm h) = f(x_i) \pm hf'(x_i) + f''(x_i)\frac{h^2}{2} + O(h^3).$$

Следовательно,

$$\frac{f(x_i + h) - f(x_i - h)}{2h} = f'(x_i) + O(h^2).$$

Аналогично получаем лево- и правосторонние формулы численного дифференцирования, точные на многочленах второй степени:

$$\begin{aligned} f'(x_i - h) &= \frac{1}{2h}[-3f(x_i - h) + 4f(x_i) - f(x_i + h)] + O(h^2), \\ f'(x_i + h) &= \frac{1}{2h}[f(x_i - h) - 4f(x_i) + 3f(x_i + h)] + O(h^2), \end{aligned}$$

Для второй производной на том же пути имеем формулу

$$\frac{1}{h^2}[f(x_i - h) - 2f(x_i) + f(x_i + h)] = f''(x_i) + O(h^2),$$

точную на кубических многочленах. Чтобы получить нецентральные аппроксимации второй производной с погрешностью $O(h^2)$, приходится использовать уже четыре точки:

$$f''(x_i) \approx \frac{1}{6h^2}[12f(x_i) - 30f(x_i + h) + 24f(x_i + 2h) - 6f(x_i + 3h)],$$
$$f''(x_i) \approx \frac{1}{6h^2}[-6f(x_i - 3h) + 24f(x_i - 2h) - 30f(x_i - h) + 12f(x_i)].$$

Эти формулы также точны на кубических многочленах.

Как следует из приведенных формул, при равном числе используемых точек точность центральных разностей оказывается на порядок по h выше, чем для нецентральных разностей.

§ 4.3. О выборе шага численного дифференцирования

При численном дифференцировании таблично заданной функции значения последней обычно известны с некоторой неустранимой погрешностью. Эта погрешность возникает, если функция определяется из измерений или вычисляется по некоторой приближенной формуле. Пусть $y_i = f(x_i) + \varepsilon_i$ и $\max_i |\varepsilon_i| \le \varepsilon$. Если мы пользуемся формулой

$$f'(x_i) \approx \frac{f(x_i + h) - f(x_i)}{h},$$

то

$$\frac{y_{i+1} - y_i}{h} = \frac{f(x_i + h) + \varepsilon_{i+1} - [f(x_i) + \varepsilon_i]}{h} = f'(x_i) + \frac{h}{2}f''(\xi) + \frac{\varepsilon_{i+1} - \varepsilon_i}{h}.$$

Следовательно,

$$\left|\frac{y_{i+1} - y_i}{h} - f'(x_i)\right| \le \varphi(h) = \frac{h}{2}M + \frac{2\varepsilon}{h}; \quad M = \max_{x_i \le x \le x_{i+1}} |f''(x)|.$$

Для малости погрешности необходима малость h, но при уменьшении h растет второе слагаемое в функции φ. Из уравнения $\varphi'(h) = 0$ получаем точку экстремума $h_e = 2\sqrt{\varepsilon/M}$, где $\varphi(h_e) = 2\sqrt{M\varepsilon}$. Так как $\varepsilon \ge const \cdot 2^{-t}$, где t – число разрядов компьютера, то мы можем получить значение $f'(x_i)$ в лучшем случае с половиной верных разрядов.

При использовании формул более высокого порядка точности ситуация не улучшается. Пусть, например:

$$f'(x_i) \approx \frac{f(x_i + h) - f(x_i - h)}{2h}.$$

Действуя аналогично, получаем:

$$\left|\frac{y_{i+1}-y_{i-1}}{2h}-f'(x_i)\right|\le\psi(h)=\frac{h^2}{6}M_3+\frac{\varepsilon}{h},\quad M_3=\max_{|x-x_i|\le h}|f'''(x)|.$$

Функция ψ достигает минимума по h при $h=\sqrt[3]{3\varepsilon/M_3}$. Таким образом, здесь мы можем получить значение $f'(x_i)$ в лучшем случае лишь с третью верных разрядов.

В ряде случаев до применения формул численного дифференцирования целесообразно провести предварительное сглаживание сеточной функции. Простейшим примером такого сглаживания является использование формулы осреднения $\tilde{y}_i=(y_{i-1}+4y_i+y_{i+1})/6$. Могут оказаться эффективными также методы, основанные на идеях математической статистики или использующие идеи регуляризации.

§ 4.4. Простейшие квадратурные формулы

При рассмотрении формул численного интегрирования мы ограничимся методами приближенного вычисления одномерных интегралов. Такие формулы называются *квадратурными*. Точность вычисления интегралов может быть увеличена за счет повышения порядка точности квадратур и/или за счет разбиения отрезка интегрирования на части. В ряде случаев целесообразно использовать адаптивные квадратурные формулы, учитывающие особенности поведения подинтегральной функции.

Пусть требуется вычислить интеграл

$$I=\int_a^b f(x)\,dx. \tag{4.3}$$

Простейшим способом приближенного вычисления этого интеграла является замена площади под кривой f на $[a,b]$ площадью прямоугольника $f(\xi)(b-a)$, где $a\le\xi\le b$. Естественно взять в качестве ξ центральную точку отрезка интегрирования. Тогда получим *формулу прямоугольников*

$$I\approx(b-a)f\Big(\frac{a+b}{2}\Big).$$

Замена площади под кривой f площадью трапеции высоты $b-a$ с основаниями $f(a)$ и $f(b)$ дает *формулу трапеций*

$$I\approx\frac{b-a}{2}[f(a)+f(b)].$$

Если теперь отрезок интегрирования $[a,b]$ разбить на N равных частей точками $x_i=a+ih$, $h=(b-a)/N$, $i=0,1,\ldots,N$ и к каждой из них применить формулы

прямоугольников и трапеций, то получим *составные* формулы прямоугольников:

$$I = \sum_{i=0}^{N-1} \int_{x_i}^{x_{i+1}} f(x)\,dx \approx h \sum_{i=0}^{N-1} f(x_{i+1/2}); \quad x_{i+1/2} = (x_i + x_{i+1})/2 \tag{4.4}$$

и трапеций

$$I = \sum_{i=0}^{N-1} \int_{x_i}^{x_{i+1}} f(x)\,dx \approx \frac{h}{2} \sum_{i=0}^{N-1} [f(x_i) + f(x_{i+1})]. \tag{4.5}$$

Отметим, что использование этих формул означает замену подинтегральной функции f ступенчатой функцией и кусочно-линейной соответственно.

Более сложные квадратурные формулы, так же как и формулы численного дифференцирования, строятся при помощи аппарата интерполирования или методом неопределенных коэффициентов.

§ 4.5. Формулы Ньютона-Котеса

В общем случае для приближенного вычисления интеграла (4.3) используются квадратурные формулы вида

$$I = \int_a^b f(x)\,dx \approx \sum_{i=0}^{n} c_i f(x_i), \tag{4.6}$$

где c_i – *коэффициенты* а x_i – *узлы* квадратурной формулы. Разность

$$R_n(f) = \int_a^b f(x)\,dx - \sum_{i=0}^{n} c_i f(x_i)$$

называется *погрешностью квадратурной формулы*. Погрешность зависит как от расположения узлов, так и от выбора коэффициентов.

Если для погрешности составной квадратурной формулы:

$$R_n^N(f) = \sum_{i=0}^{N-1} \Big[\int_{x_i}^{x_{i+1}} f(x)\,dx - \sum_{j=0}^{n} c_j f(x_{ij}) \Big]; \quad x_{ij} = x_i + jh/n, j = 0, 1, \ldots, n$$

имеет место оценка

$$|R_n^N(f)| \le const \cdot N \Big(\frac{b-a}{N}\Big)^k = const \cdot (b-a) h^{k-1},$$

то число $k-1$ называется *порядком точности квадратурной формулы*.

Заменив подинтегральную функцию f в (4.6) интерполяционным многочленом Лагранжа с узлами $a = x_0 < x_1 < \ldots < x_n = b$, получим квадратурную формулу интерполяционного типа

$$I = \int_a^b f(x)\,dx \approx \int_a^b L_n(x)\,dx = \sum_{i=0}^{n} f(x_i) \int_a^b l_i(x)\,dx = \sum_{i=0}^{n} c_i f(x_i), \tag{4.7}$$

где фундаментальные многочлены Лагранжа l_i имеют вид:

$$l_i(x) = \frac{\omega_n(x)}{(x - x_i)\omega_n'(x_i)}; \quad \omega_n(x) = \prod_{i=0}^{n}(x - x_i).$$

В формуле (4.7) коэффициенты $c_i = \int_a^b l_i(x)dx$ не зависят от подинтегральной функции f, а зависят только от положения узлов интерполяции x_i. Квадратурная формула интерполяционного типа, построенная по $n+1$ узлу, точна на многочленах степени n, т. е. $R_n(f) = 0$, так как в этом предположении $f(x) \equiv L_n(x)$. Справедливо и обратное: квадратурная формула вида (4.6), точная на многочленах степени n, является квадратурной формулой интерполяционного типа. Действительно в этом случае в силу свойств фундаментальных многочленов Лагранжа l_i получаем

$$\int_a^b l_i(x)\,dx = \sum_{j=0}^{n} c_j l_i(x_j) = \sum_{j=0}^{n} c_j \delta_{ij} = c_i.$$

Если функция $f \in C^{n+1}[a,b]$, то $f(x) = L_n(x) + \omega_n(x) f^{(n+1)}(\xi)/(n+1)!$. В этом случае погрешность квадратурной формулы интерполяционного типа можно представить в виде:

$$R_n(f) = \int_a^b \omega_n(x) \frac{f^{(n+1)}(\xi)}{(n+1)!}\,dx. \tag{4.8}$$

При равномерном расположении узлов на отрезке интегрирования квадратурные формулы интерполяционного типа (4.7) называются *квадратурными формулами Ньютона-Котеса*.

Приведенные выше квадратурные формулы прямоугольников и трапеций являются частными случаями формул Ньютона-Котеса. Рассмотрим случай квадратичного многочлена Лагранжа. Используя обозначение $(x - x_i)/h = t$, получаем

$$L_{i,2}(x) = f(x_i)(2t-1)(t-1) + f(x_{i+1/2})4t(1-t) + f(x_{i+1})t(2t-1).$$

Интегрирование дает нам *квадратурную формулу Симпсона*

$$\begin{aligned} &\int_{x_i}^{x_{i+1}} f(x)\,dx \approx \int_{x_i}^{x_{i+1}} L_{i,2}(x)\,dx = h\int_0^1 L_{i,2}(x_i + ht)\,dt = \\ = \;& h\Big[f(x_i)\int_0^1 (2t-1)(t-1)dt + f(x_{i+1/2})\int_0^1 4t(1-t)dt + \\ &+ f(x_{i+1})\int_0^1 t(2t-1)dt\Big] = \frac{h}{6}[f(x_i) + 4f(x_{i+1/2}) + f(x_{i+1})], \end{aligned}$$

точную на кубических многочленах.

Суммируя теперь по i, получаем составную формулу Симпсона

$$\int_a^b f(x)\,dx \approx \frac{h}{6}\sum_{i=0}^{N-1}[f(x_i)+4f(x_{i+1/2})+f(x_{i+1})].$$

Отметим, что при $n = 1, 2, \ldots, 7, 9$ все коэффициенты c_i в формулах Ньютона-Котеса вида (4.7) положительны. При $n = 8$ и $n \geq 10$ среди них имеются как положительные, так и отрицательные. По этой причине формулы Ньютона-Котеса не рекомендуется применять при больших n.

§ 4.6. Оценки погрешности квадратурных формул

Оценим погрешность формулы прямоугольников на $[x_i, x_{i+1}]$, используя разложение функции f по формуле Тейлора:

$$f(x) = f(x_{i+1/2}) + f'(x_{i+1/2})(x - x_{i+1/2}) + \frac{1}{2}f''(\xi)(x - x_{i+1/2})^2; \quad \xi \in [x_i, x_{i+1}].$$

Так как

$$R_{i,0} = \int_{x_i}^{x_{i+1}} f(x)\,dx - hf(x_{i+1/2}) = \frac{1}{2}\int_{x_i}^{x_{i+1}} f''(\xi)(x - x_{i+1/2})^2 dx,$$

то получаем оценку

$$|R_{i,0}| \leq \frac{M_{i,2}}{2}\int_{x_i}^{x_{i+1}}(x - x_{i+1/2})^2 dx = M_{i,2}\frac{h^3}{24}; \quad M_{i,2} = \max_{x_i \leq x \leq x_{i+1}} |f''(x)|.$$

Эта оценка является неулучшаемой. Она достигается, например, на функции $f(x) = (x - x_{i+1/2})^2$.

Для составной формулы прямоугольников (4.4) получаем оценку

$$|R_0^N| = \sum_{i=0}^{N-1} |R_{i,0}| \leq M_2\frac{(b-a)h^2}{24}; \quad M_2 = \max_{a \leq x \leq b} |f''(x)|.$$

Для оценки погрешности формулы трапеций на отрезке $[x_i, x_{i+1}]$ воспользуемся формулой (4.8):

$$R_{i,1} = \frac{1}{2}\int_{x_i}^{x_{i+1}} f''(\xi)(x - x_i)(x - x_{i+1})\,dx,$$

откуда следует оценка

$$|R_{i,1}| \leq M_{i,2}\frac{h^3}{12}.$$

Эта оценка также является точной. Равенство в ней достигается, например, на функции $f(x) = (x - x_i)^2$.

Выпишем оценку для составной формулы трапеций:

$$|R_1^N| = \sum_{i=0}^{N-1} |R_{i,1}| \le M_2 \frac{(b-a)h^2}{12}.$$

Таким образом, обе формулы прямоугольников и трапеций имеют второй порядок точности по h, но постоянная в формуле прямоугольников в два раза меньше.

Чтобы оценить погрешность формулы Симпсона на отрезке $[x_i, x_{i+1}]$, рассмотрим кубический многочлен:

$$\begin{aligned} H_{i,3}(x) &= f(x_i) + f[x_i, x_{i+1}](x - x_i) + f[x_i, x_{i+1}, x_{i+1/2}](x - x_i)(x - x_{i+1}) \\ &\quad + f[x_i, x_{i+1}, x_{i+1/2}, x_{i+1/2}](x - x_i)(x - x_{i+1})(x - x_{i+1/2}). \end{aligned}$$

Так как формула Симпсона точна на кубических многочленах, то

$$\begin{aligned} R_{i,2} &= \int_{x_i}^{x_{i+1}} f(x)\,dx - \frac{h}{6}[f(x_i) + 4f(x_{i+1/2}) + f(x_{i+1})] = \\ &= \int_{x_i}^{x_{i+1}} [f(x) - H_{i,3}(x)]\,dx = \\ &= \int_{x_i}^{x_{i+1}} f[x_i, x_{i+1}, x_{i+1/2}, x_{i+1/2}, x](x - x_i)(x - x_{i+1})(x - x_{i+1/2})^2\,dx = \\ &= \int_{x_i}^{x_{i+1}} \frac{f^{(4)}(\xi)}{4!}(x - x_i)(x - x_{i+1})(x - x_{i+1/2})^2\,dx, \quad \xi \in [x_i, x_{i+1}]. \end{aligned}$$

Отсюда при обозначении $M_{i,4} = \max_{x_i \le x \le x_{i+1}} |f^{(4)}(x)|$ получаем

$$|R_{i,2}| \le \frac{M_{i,4}}{4!} \int_{x_i}^{x_{i+1}} (x - x_i)(x_{i+1} - x)(x - x_{i+1/2})^2\,dx = M_{i,4}\frac{h^5}{2880}.$$

Равенство достигается на функции $f(x) = (x - x_i)^4$. Проводя теперь суммирование по i, находим оценку для составной формулы Симпсона:

$$|R_2^N| = \sum_{i=0}^{N-1} |R_{i,2}| \le M_4 \frac{(b-a)h^4}{2880}; \quad M_4 = \max_{a \le x \le b} |f^{(4)}(x)|.$$

Таким образом, формула Симпсона, имеющая четвертый порядок точности по h, существенно точнее, чем формулы прямоугольников и трапеций.

§ 4.7. Метод неопределенных коэффициентов

Пусть требуется построить квадратурную формулу

$$\int_a^b f(x)\,dx = \sum_{i=0}^{n} c_i f(x_i) + R_n(f)$$

с фиксированными узлами, точную на многочленах наиболее высокой степени. Это означает, что должны выполняться равенства:

$$R_n(x^j) = \frac{b^{j+1} - a^{j+1}}{j+1} - \sum_{i=0}^{n} c_i x_i^j = 0; \quad j = 0, 1, \ldots, m.$$

при возможно большем значении m.

В качестве примера рассмотрим квадратурную формулу

$$\int_a^b f(x)\,dx = c_0 f(a) + c_1 f\Big(\frac{a+b}{2}\Big) + c_2 f(b) + R_2(f).$$

Для нахождения коэффициентов c_i имеем систему линейных уравнений:

$$\frac{b^{j+1} - a^{j+1}}{j+1} = c_0 a^j + c_1 \Big(\frac{a+b}{2}\Big)^j + c_2 b^j; \quad j = 0, 1, ...m.$$

Из первых трех уравнений находим уже известную нам формулу Симпсона:

$$\int_a^b f(x)\,dx \approx \frac{b-a}{6}\Big[f(a) + 4f\Big(\frac{a+b}{2}\Big) + f(b)\Big].$$

Четвертое уравнение выполняется автоматически, т. е. мы получили квадратуру, точную на кубических многочленах. Так как коэффициенты c_i уже определены, то пятое уравнение позволяет найти погрешность квадратурной формулы:

$$R_2(f) = -\frac{(b-a)^5}{2880} f^{(4)}(\xi), \quad \xi \in [a, b].$$

§ 4.8. Квадратурные формулы Гаусса

Рассмотрим теперь квадратурную формулу

$$\int_a^b f(x)\,dx = \sum_{i=1}^{n} c_i f(x_i) + R_n(f), \tag{4.9}$$

в которой как коэффициенты так и узлы выбираются из условия ее точности на многочленах максимально высокой степени. Квадратурные формулы такого типа называются *квадратурными формулами Гаусса*. Они имеют неравномерное расположение узлов. Так как у нас $2n$ неизвестных, то возьмем $2n$ условий точности нашей квадратурной формулы на мономах

$$R_n(x^j) = \frac{b^{j+1} - a^{j+1}}{j+1} - \sum_{i=1}^{n} c_i x_i^j = 0; \quad j = 0, 1, \ldots, 2n-1. \tag{4.10}$$

Это уже нелинейная система алгебраических уравнений, которая однако имеет единственное решение [3]. Рассмотрим два частных случая.

При $n = 1$ система (4.10) принимает вид:

$$\begin{cases} c_1 = b - a; \\ c_1 x_1 = \frac{1}{2}(b^2 - a^2), \end{cases}$$

откуда $x_1 = (a+b)/2$. Это дает уже рассмотренную нами формулу прямоугольников, которая, таким образом, является формулой Гаусса.

При $n = 2$ система (4.10) записывается в виде:

$$\begin{cases} c_1 + c_2 = b - a; \\ c_1 x_1 + c_2 x_2 = \frac{1}{2}(b^2 - a^2); \\ c_1 x_1^2 + c_2 x_2^2 = \frac{1}{3}(b^3 - a^3); \\ c_1 x_1^3 + c_2 x_2^3 = \frac{1}{4}(b^4 - a^4). \end{cases}$$

Отсюда

$$c_1 = c_2 = \frac{b-a}{2}; \quad x_{1,2} = \frac{a+b}{2} \mp \frac{\sqrt{3}}{3}\frac{b-a}{2}.$$

Это дает нам квадратурную формулу, точную на кубических многочленах.

Решение системы (4.10) в общем случае затруднительно. Однако на помощь нам приходит следующее утверждение.

Теорема 4.1. *Квадратурная формула* (4.9) *точна на многочленах степени* $2n-1$ *тогда и только тогда, когда выполняются следующие два условия:*

1) многочлен $\omega_n(x) = (x - x_1)(x - x_2)\dots(x - x_n)$, *составленный по узлам квадратурной формулы* (4.9), *ортогонален любому многочлену* q *степени* $n-1$, *т. е.*

$$\int_a^b \omega_n(x) q(x)\, dx = 0; \tag{4.11}$$

2) формула (4.9) *является квадратурной формулой интерполяционного типа, т. е.*

$$c_i = \int_a^b l_i(x)\, dx; \quad i = 1, 2, \dots, n. \tag{4.12}$$

Доказательство. *Необходимость.* Пусть формула (4.9) точна на многочленах степени $2n-1$. Тогда она будет точна и на многочлене $\omega_n q$, так как он имеет

степень не выше $2n-1$. Поэтому

$$\int_a^b \omega_n(x)q(x)\,dx = \sum_{i=1}^{n} c_i\omega_n(x_i)q(x_i) = 0,$$

т. е. условие 1) выполнено. Справедливость условия 2) уже отмечалась нами в разделе 4.5.

Достаточность. Пусть f – произвольный многочлен степени $2n-1$. Тогда его всегда можно представить в виде:

$$f(x) = \omega_n(x)q(x) + r(x),$$

где r – многочлен степени не выше $n-1$. Следовательно,

$$\begin{aligned}\int_a^b f(x)\,dx &= \int_a^b \omega_n(x)q(x)\,dx + \int_a^b r(x)\,dx = \sum_{i=1}^{n} c_i r(x_i) = \\ &= \sum_{i=1}^{n} c_i[f(x_i) - \omega_n(x_i)q(x_i)] = \sum_{i=1}^{n} c_i f(x_i).\end{aligned}$$

Теорема доказана.

Положим

$$\omega_n(x) = \frac{n!}{(2n)!}\frac{d^n}{dx^n}[(x-a)^n(x-b)^n].$$

Используя последовательное интегрирование по частям, нетрудно показать, что условие ортогональности (4.11) выполняется. Последовательное применение теоремы Ролля показывает, что все корни уравнения $\omega_n(x) = 0$ действительны, различны и заключены в интервале (a,b). Таким образом, их действительно можно использовать в качестве узлов интерполяции и полученная при этом квадратурная формула будет удовлетворять поставленным условиям.

Вычисление коэффициентов квадратуры Гаусса (4.9) по формуле (4.12) приводит к следующему результату:

$$c_i = \frac{(n!)^4(b-a)^{2n+1}}{[(2n)!]^2(x_i-a)(b-x_i)[\omega_n'(x_i)]^2}; \quad i = 1, 2, \ldots, n.$$

Остаточный член в квадратуре Гаусса (4.9) имеет вид:

$$R_n(f) = \frac{f^{(2n)}(\xi)}{(2n)!}\int_a^b \omega_n^2(x)\,dx = \frac{(b-a)^{2n+1}(n!)^4}{[(2n)!]^3(2n+1)}f^{(2n)}(\xi), \quad \xi \in [a,b].$$

Квадратурные формулы Гаусса целесообразно применять при $n > 2$ для приближенного вычисления интегралов от функций, имеющих высокую гладкость. Ввиду высокой точности и экономичности такие квадратурные формулы весьма эффективны, когда значения функции при большом числе значений аргумента получить затруднительно.

§ 4.9. Формулы Гаусса-Чебышева

Рассмотрим формулу Гаусса для вычисления интеграла вида

$$\int_{-1}^{1} \frac{f(x)dx}{\sqrt{1-x^2}} = \sum_{i=1}^{n} c_i f(x_i) + R_n(f). \tag{4.13}$$

Здесь функцию $p(x) = (1-x^2)^{-1/2}$ принято называть весовой функцией.

Узлы x_i должны быть корнями многочлена ω_n, который ортогонален с весом $(1-x^2)^{-1/2}$ произвольному многочлену q степени не выше $n-1$, т. е.

$$\int_{-1}^{1} \frac{\omega_n(x)x^k}{\sqrt{1-x^2}} = 0; \quad k = 0, 1, \ldots, n-1. \tag{4.14}$$

Докажем, что таким свойством обладает многочлен $\omega_n(x) = 2^{-n+1}T_n(x)$, где многочлен Чебышева $T_n(x) = \cos(n \arccos(x))$. Полагая $x = \cos\varphi$, имеем

$$\int_{-1}^{1} \frac{\omega_n(x)x^k}{\sqrt{1-x^2}}dx = \frac{1}{2^{n-1}} \int_0^{\pi} \cos n\varphi \cos^k \varphi \, d\varphi.$$

Так как

$$\cos^k \varphi = a_0 + a_1 \cos\varphi + \ldots + a_k \cos k\varphi$$

и

$$\int_0^{\pi} \cos n\varphi \cos l\varphi \, d\varphi = 0 \quad \text{при} \quad l \le n-1,$$

то равенства (4.14) выполняются.

Таким образом, узлами квадратурной формулы (4.13) являются корни многочлена Чебышева:

$$x_i = cos\frac{(2i-1)\pi}{2n}; \quad i = 1, 2, \ldots, n.$$

Коэффициенты c_i вычисляются по формуле

$$c_i = \int_{-1}^{1} \frac{l_i(x)}{\sqrt{1-x^2}}dx = \int_{-1}^{1} \frac{T_n(x)dx}{(x-x_i)T_n'(x_i)\sqrt{1-x^2}} = \frac{\pi}{n}; \quad i = 1, 2, \ldots, n.$$

Остаточный член имеет вид:

$$R_n(f) = \frac{f^{(2n)}(\xi)}{(2n)!} \int_{-1}^{1} \frac{T_n^2(x)dx}{\sqrt{1-x^2}} = \frac{\pi}{(2n)!2^{2n-1}} f^{(2n)}(\xi), \quad \xi \in [-1, 1].$$

Полученная формула носит название *формулы Эрмита*.

§ 4.10. Правило Рунге практической оценки погрешности

Пусть требуется вычислить интеграл (4.3) с заданной точностью ε. Если мы пользуемся составной квадратурной формулой , то дело сводится к выбору такого шага $h = (b-a)/N$, чтобы погрешность квадратурной формулы удовлетворяла условию $|R_n^N(f)| \le \varepsilon$.

Например, в случае составной формулы Симпсона можно потребовать, чтобы

$$M_4 \frac{(b-a)h^4}{2880} \le \varepsilon \quad \text{или} \quad h < \Big(\frac{2880\varepsilon}{(b-a)M_4}\Big)^{1/4}.$$

Однако на практике этой априорной оценкой воспользоваться затруднительно, так как величина $M_4 = \max_{x\in[a,b]} |f^{(4)}(x)|$ обычно неизвестна. Для того, чтобы правильно выбрать шаг h, нужно оценивать погрешность после проведения расчета, т. е. апостериорную погрешность. Одним из практических приемов выбора шага h на основе оценивания апостериорной погрешности является *метод Рунге*.

Пусть

$$I_i = \int_{x_i}^{x_{i+1}} f(x)\,dx = I_{h,i} + R_i,$$

где $I_{h,i}$ – некоторая квадратурная формула, $R_i = C_i h^p + O(h^{p+1})$ – погрешность квадратурной формулы а p – порядок точности этой квадратуры.

Вычислим тот же интеграл, используя шаг $h/2$. Тогда

$$I_i = I_{h/2,i} + C_i\Big(\frac{h}{2}\Big)^p + O(h^{p+1}).$$

Из этих двух равенств получаем *формулу Рунге* для уточнения найденного значения интеграла

$$I_i = I_{h/2,i} + \frac{I_{h/2,i} - I_{h,i}}{2^p - 1} + O(h^{p+1}). \tag{4.15}$$

Если теперь для главного члена погрешности в этой формулы справедлива оценка

$$\frac{|I_{h/2,i} - I_{h,i}|}{2^p - 1} < \varepsilon, \tag{4.16}$$

то вычисления на отрезке $[x_i, x_{i+1}]$ прекращаем. В противном случае опять уменьшаем шаг в два раза и повторяем вычисления до тех пор, пока нужное нам неравенство не будет выполнено. Таким образом, алгоритм сам определяет шаг сетки, исходя из заданной точности. Более того, мы можем получить сетку, учитывающую поведение подинтегральной функции. На участках больших градиентов она

будет мельчиться и иметь крупный шаг на участках плавного изменения функции. Алгоритмы такого типа называются *адаптивными квадратурными формулами*. Оптимальной будет сетка, для которой погрешность на всех подинтервалах примерно одинакова.

Следует указать на некоторые «подводные камни» метода Рунге. Неравенство (4.16) может не выполняться по следующим причинам:

1) слишком велико h и, следовательно, велико слагаемое $O(h^{p+1})$;

2) слишком мало h и сказываются ошибки округления ЭВМ;

3) постоянная $C_i \approx 0$.

Правило Рунге можно использовать и для уточнения вычисленной величины I_i. Из (4.15) следует, что более точным будет значение

$$I_i^* = \frac{2^p I_{h/2,i} - I_i}{2^p - 1}.$$

Этот прием называется *экстраполяцией по Ричардсону*. Пусть, например, мы пользуемся методом трапеций и, следовательно, $p = 2$. Тогда

$$\begin{aligned} I_i^* &= \frac{1}{3}(4I_{h/2,i} - I_i) = \frac{1}{3}\Big[4\frac{h}{2}\Big(\frac{1}{2}f(x_i) + f(x_{i+1/2}) + \frac{1}{2}f(x_{i+1})\Big) - \\ &\quad -h\Big(\frac{1}{2}f(x_i) + \frac{1}{2}f(x_{i+1})\Big)\Big] = \frac{h}{6}[f(x_i) + 4f(x_{i+1/2}) + f(x_{i+1})], \end{aligned}$$

т. е. мы получили формулу Симпсона.

§ 4.11. Задачи

4.1. Пусть $f(x) = \cos \pi x$. Используя значения f в точках $x = 0,25; 0,5; 0,75$, найдите приближенное значение $f''(0,5)$. Оцените погрешность полученного приближения.

4.2. Машина движется по прямолинейной дороге. Ее положение в разные моменты времени указано в приводимой таблице. Какую скорость имеет машина в указанные в таблице моменты времени? Воспользуйтесь формулой численного дифференцирования по трем точкам.

Время в секундах	0	3	5	8	10	13
Расстояние	0	225	383	623	742	993

4.3. Методом неопределенных коэффициентов, по точкам $x_i + kh$, $k = 0, 1, 2, 3$, найдите формулы численного дифференцирования для $f'''(x_i)$ и $f'''(x_i + 3h)$.

4.4. Каков оптимальный шаг численного дифференцирования для формулы

$$f''(x_i) \approx \frac{1}{h^2}[f(x_i - h) - 2f(x_i) + f(x_i + h)],$$

если значения функции f вычисляются с точностью $\varepsilon = 10^{-2}$?

4.5. Требуется вычислить интеграл $\int_0^2 \frac{1}{x+4}\,dx$, используя составные формулы прямоугольников, трапеций и Симпсона. Какой шаг h следует выбрать для каждой их этих формул, чтобы получить значение интеграла с точностью $\varepsilon = 10^{-5}$?

4.6. Найдите приближенные значения следующих интегралов по формулам трапеций и Симпсона и сравните полученные результаты:

а) $\int_{0,5}^{1} x^4\,dx$; б) $\int_1^{1,6} \frac{2x}{x^2-4}\,dx$;

в) $\int_0^{0,5} \frac{2}{x-4}\,dx$; г) $\int_1^{1,6} \frac{2}{x^2-4}\,dx$;

д) $\int_1^{1,5} x^2 \ln x\,dx$; е) $\int_0^{\pi/4} x \sin x\,dx$;

ж) $\int_0^1 x^2 e^{-x}\,dx$; з) $\int_0^{\pi/4} e^{3x} \sin 2x\,dx$.

Оцените погрешность при вычисление интегралов и сравните ее с получаемой фактически.

4.7. Получите оценку погрешности правила трех восьмых

$$\int_{x_i}^{x_{i+1}} f(x)\,dx = \frac{3h}{8}[f(x_i) + 3f(x_i + h/3) + 3f(x_i + 2h/3) + f(x_{i+1})] + R_{3,i}(f).$$

4.8. Каков порядок точности квадратурной формулы

$$\int_{-1}^{1} f(x)\,dx = f\Big(-\frac{\sqrt{3}}{3}\Big) + f\Big(\frac{\sqrt{3}}{3}\Big)?$$

4.9. Найдите c_0, c_1 и x_1 такие, что квадратурная формула

$$\int_0^1 f(x)\,dx = c_0 f(0) + c_1 f(x_1)$$

имеет наибольший возможный порядок точности.

4.10. Найдите x_0, x_1 и c_1 такие, что квадратурная формула

$$\int_0^1 f(x)\,dx = \frac{1}{2} f(x_0) + c_1 f(x_1)$$

имеет наибольший возможный порядок точности.

4.11. Покажите, что формула

$$\int_{-1}^{1} \frac{f(x)}{\sqrt{1-x^2}}\,dx = \frac{\pi}{3}\Big[f\Big(-\frac{\sqrt{3}}{2}\Big) + f(0) + f\Big(\frac{\sqrt{3}}{2}\Big)\Big]$$

точна для многочленов пятой степени.

4.12. Для приближенного вычисления интеграла $\int_{-2}^{2} f(x)\,dx$ постройте квадратурную формулу с узлами $-1, 0, 1$, используя интерполяционный многочлен.

4.13. Длина кривой вычисляется по правилу $\int_a^b \sqrt{1+[f'(x)]^2}\,dx$, где f – функция, график которой дает кривую на отрезке $[a, b]$. Вычислить с точностью до 10^{-6} длину кривой эллипса в случае уравнения $4x^2 + y^2 = 1$.

4.14. Используя правило Рунге, постройте адаптивные квадратурные формулы и вычислите интегралы

$$\int_{0,1}^{2} \sin\frac{1}{x}\,dx \quad \text{и} \quad \int_{0,1}^{2} \cos\frac{1}{x}\,dx$$

с точностью 10^{-3}.

Глава 5
Решение систем линейных уравнений

Системы линейных уравнений появляются почти в каждой области прикладной математики. В некоторых случаях эти системы уравнений непосредственно составляют задачу, которую необходимо решить. В других случаях задача сводится к такой системе. Например, линейную систему приходится решать при построении интерполяционного или сглаживающего сплайна (гл. 2 и 3). При использовании разностных методов для решения линейных дифференциальных уравнений также требуется решать системы линейных уравнений. Существует множество других задач, сводящихся к решению систем линейных уравнений.

§ 5.1. Методы решения систем линейных уравнений

Рассмотрим систему линейных алгебраических уравнений

$$\mathbf{A}\mathbf{x} = \mathbf{b}, \tag{5.1}$$

где $\mathbf{A} = \{a_{ij}\}$, $i, j = 1, 2, \ldots, n$, – невырожденная вещественная квадратная матрица, $\mathbf{b} = (b_1, \ldots, b_n)^T$ – заданный вектор-столбец, а $\mathbf{x} = (x_1, \ldots, x_n)^T$ – столбец неизвестных.

Если $\tilde{\mathbf{x}}$ и $\mathbf{x}^*$ – приближенное и точное решения системы (1.1) соответственно, то вектор $\mathbf{w} = \tilde{\mathbf{x}} - \mathbf{x}^*$ называют *погрешностью* приближенного решения а вектор $\mathbf{r} = \mathbf{b} - \mathbf{A}\tilde{\mathbf{x}}$ – *невязкой*.

Если $\det(\mathbf{A}) = 0$, то матрица системы (1.1) является вырожденной. Ее решение может не существовать или быть неединственным. Существуют специальные методы решения систем с такими матрицами. Однако мы не будем их рассматривать, предполагая $\det(\mathbf{A}) \neq 0$.

Определение 1.1. *Будем считать, что задача* (1.1) *поставлена корректно, если ее решение существует, оно единственно и непрерывно зависит от входных данных.*

Под непрерывной зависимостью решения от входных данных понимается незначительное изменение решения при малом возмущении входных данных – матрицы и правой части системы (1.1). Отметим, что матрицы могут быть классифицированы как плотные и разреженные, произвольные и специальные и т. д.

Методы решения систем вида (1.1) делятся на прямые и итерационные. В первом случае предполагается, что решение может быть получено за конечное число арифметических операций. Сюда относится метод Гаусса и его разновидности, ориентированные на специальные матрицы (симметрические, положительно определенные, ленточные и др.). Прямые методы, использующие ортогональные преобразования (методы вращений, ортогонализации, отражений и др.), существенно более устойчивы по отношению к ошибкам округления, но и более трудоемки.

В итерационных методах решение получается как предел бесконечной последовательности, которая на практике обрывается при достижении заданной точности. Если итерационные методы сходятся быстро, то они обычно предпочтительнее прямых методов. Объем вычислений в них на одну итерацию равен $O(n^2)$, тогда как гауссово исключение требует $O(n^3)$. Поэтому при числе итераций $k < n$ общие затраты будут меньше. К тому же итерационные методы имеют тенденцию быть самокорректирующимися, и, следовательно, минимизируют ошибки округления. Каждая итерация может рассматриваться как новое начальное приближение. Это соображение может оказаться важным оправданием дополнительных вычислений.

В практических задачах матрица системы (1.1) часто содержит много нулей. Как правило, итерационные методы позволяют сохранять эти нули и экономить вычисления. Можно также существенно уменьшить объем используемой памяти ЭВМ, вычисляя коэффициенты каждого уравнения тогда, когда в них возникает необходимость. Это прежде всего относится к системам, возникающим при решении уравнений в частных производных и многомерной интерполяции сплайнами. Важную роль для ускорения сходимости итерационных методов играют предобуславливатели.

Не все методы подходят для компьютерных вычислений. Например, классический метод Крамера, основанный на вычислении детерминантов порядка n, требует для своей реализации $n!n$ арифметических операций. Уже при $n = 30$ такое число операций недоступно для современных ЭВМ. Сравнение методов обычно проводят по трем характеристикам: объему используемой памяти ЭВМ, количеству арифметических операций, требующихся для реализации метода, и устойчивости метода по отношению к ошибкам округления. Важную роль также играет возможность распараллеливания вычислений. Это относится как к прямым так и итерационным методам.

§ 5.2. Нормы векторов и матриц

В конечномерном векторном пространстве $\mathbb{R}^n$ функция $\|\cdot\|\colon \mathbb{R}^n \to \mathbb{R}$, называемая векторной нормой, удовлетворяет следующим трем аксиомам. Для всяких

векторов $\mathbf{x}, \mathbf{y} \in \mathbb{R}^n$ и произвольного числа $c \in \mathbb{R}$ имеем:

1. $\|\mathbf{x}\| \geq 0$ и $\|\mathbf{x}\| = 0$, если и только если $\mathbf{x} = \mathbf{0}$;

2. $\|c\mathbf{x}\| = |c| \cdot \|\mathbf{x}\|$;

3. $\|\mathbf{x} + \mathbf{y}\| \leq \|\mathbf{x}\| + \|\mathbf{y}\|$.

Норма матрицы $\mathbf{A}$ определяется посредством векторной нормы по правилу

$$\|\mathbf{A}\| = \sup_{\mathbf{x} \neq \mathbf{0}} \frac{\|\mathbf{A}\mathbf{x}\|}{\|\mathbf{x}\|}. \tag{5.2}$$

Отметим, что, так как

$$\|\mathbf{A}\| = \sup_{\mathbf{x} \neq \mathbf{0}} \frac{\|\mathbf{A}\mathbf{x}\|}{\|\mathbf{x}\|} = \sup_{\mathbf{x} \neq \mathbf{0}} \left\| \frac{1}{\|\mathbf{x}\|} \mathbf{A}\mathbf{x} \right\| = \sup_{\mathbf{x} \neq \mathbf{0}} \left\| \mathbf{A} \frac{\mathbf{x}}{\|\mathbf{x}\|} \right\| = \sup_{\|\mathbf{y}\|=1} \|\mathbf{A}\mathbf{y}\|,$$

то матричная норма может быть определена эквивалентным образом как

$$\|\mathbf{A}\| = \sup_{\|\mathbf{x}\|=1} \|\mathbf{A}\mathbf{x}\|. \tag{5.3}$$

Для целого числа $p \geq 1$ норму вектора $\mathbf{x} \in \mathbb{R}^n$ определим по правилу

$$\|\mathbf{x}\|_p = \left(\sum_{i=1}^{n} |x_i|^p \right)^{1/p}.$$

Покажем, что $\lim\limits_{p \to \infty} \|\mathbf{x}\|_p = \max\limits_i |x_i|$. Пусть максимальной по модулю компонентой вектора $\mathbf{x}$ является k-я компонента, т. е. $\max\limits_i |x_i| = |x_k|$, $1 \leq k \leq n$. Тогда получаем

$$\lim_{p \to \infty} \|\mathbf{x}\|_p = \lim_{p \to \infty} \left(\sum_{i=1}^{n} |x_i|^p \right)^{1/p} = |x_k| \lim_{p \to \infty} \left(\sum_{i=1}^{n} \left| \frac{x_i}{x_k} \right|^p \right)^{1/p} = \max_{1 \leq i \leq n} |x_i|.$$

Здесь использован тот факт, что k-е слагаемое в последней сумме равно единице, а все остальные слагаемые не превосходят единицы. Поэтому вся сумма S удовлетворяет условию $1 \leq S \leq n$ и $\lim_{p \to \infty} S^{1/p} = 1$.

Наиболее употребительными на практике являются следующие три векторные нормы:

$$\|\mathbf{x}\|_1 = \sum_{i=1}^{n} |x_i|, \quad \|\mathbf{x}\|_2 = \sqrt{\sum_{i=1}^{n} |x_i|^2}, \quad \|\mathbf{x}\|_\infty = \max_i |x_i|,$$

которым согласно (1.2) и (1.3) соответствуют матричные нормы:

$$\|\mathbf{A}\|_1 = \max_{1 \leq j \leq n} \sum_{i=1}^{n} |a_{ij}|, \ \|\mathbf{A}\|_2 = \sqrt{\max_{1 \leq i \leq n} |\lambda_i(\mathbf{A}^T\mathbf{A})|}, \ \|\mathbf{A}\|_\infty = \max_{1 \leq i \leq n} \sum_{j=1}^{n} |a_{ij}|.$$

Здесь $\lambda_i(\mathbf{A}^T\mathbf{A})$ – собственные значения матрицы $\mathbf{A}^T\mathbf{A}$, которые принято называть *сингулярными числами* матрицы A. В частности, если $\mathbf{A}$ – симметрическая матрица, то $\lambda_i(\mathbf{A}^T\mathbf{A}) = \lambda_i(\mathbf{A}^2) = |\lambda_i(\mathbf{A})|^2$. Поэтому для симметрической матрицы

$$\|\mathbf{A}\|_2 = \max_{1\le i\le n} |\lambda_i(\mathbf{A})|.$$

§ 5.3. Плохообусловленные системы

Система (1.1) считается *плохообусловленной*, если малые изменения коэффициентов матрицы $\mathbf{A}$ и/или компонент правой части $\mathbf{b}$ вызывают существенное изменение решения этой системы $\mathbf{x}$.

Рассмотрим количественные характеристики понятия плохой обусловленности системы (1.1). Числом обусловленности матрицы $\mathbf{A}$ называют величину

$$\mathrm{cond}(\mathbf{A}) = \|\mathbf{A}\| \cdot \|\mathbf{A}^{-1}\|.$$

Отметим, что это число не может быть меньше единицы, так как

$$1 = \|\mathbf{E}\| = \|\mathbf{A}\cdot\mathbf{A}^{-1}\| \le \|\mathbf{A}\| \cdot \|\mathbf{A}^{-1}\| = \mathrm{cond}(\mathbf{A}).$$

Для симметрической матрицы $\mathbf{A}$ из равенств $\mathbf{A}\mathbf{x} = \lambda\mathbf{x}$ и $\mathbf{A}^{-1}\mathbf{x} = \lambda^{-1}\mathbf{x}$ имеем

$$\|\mathbf{A}\|_2 = \max_i |\lambda_i|, \quad \|\mathbf{A}^{-1}\|_2 = \frac{1}{\min_i |\lambda_i|},$$

т. е. в этом случае

$$\mathrm{cond}_2(\mathbf{A}) = \|\mathbf{A}\|_2\|\mathbf{A}^{-1}\|_2 = \frac{\max_i |\lambda_i|}{\min_i |\lambda_i|} \ge 1.$$

Обычно матрица $\mathbf{A}$ называется плохообусловленной, если ее число обусловленности порядка тысяч. Системы линейных уравнений с такими матрицами плохо поддаются решению. Если $\mathbf{A}$ – вырожденная матрица, то $\mathrm{cond}(\mathbf{A}) = \infty$.

Сравним число обусловленности матрицы с ее детерминантом. Пусть дана диагональная матрица $\mathbf{A} = \varepsilon\mathbf{E}$ размерности $n \times n$, где $\varepsilon > 0$ – малое число и $\mathbf{E}$ – единичная матрица. Здесь $\det(\mathbf{A}) = \varepsilon^n$ весьма мал, тогда как $\mathrm{cond}(\mathbf{A}) = 1$ при любом n.

Рассмотрим теперь матрицу $\mathbf{A}$ и обратную к ней $\mathbf{A}^{-1}$ вида

$$\mathbf{A} = \begin{pmatrix} 1 & -1 & -1 & \dots & -1 \\ 0 & 1 & -1 & \dots & -1 \\ \vdots & \vdots & \ddots & \ddots & \vdots \\ 0 & 0 & \dots & 1 & -1 \\ 0 & 0 & \dots & 0 & 1 \end{pmatrix}; \quad \mathbf{A}^{-1} = \begin{pmatrix} 1 & 1 & 2 & \dots & 2^{n-2} \\ 0 & 1 & 1 & \dots & 2^{n-3} \\ \vdots & \vdots & \ddots & \ddots & \vdots \\ 0 & 0 & \dots & 1 & 1 \\ 0 & 0 & \dots & 0 & 1 \end{pmatrix}.$$

Детерминант $\det(\mathbf{A}) = 1$. Используя первую норму, получаем:

$$\|\mathbf{A}\|_1 = n, \quad \|\mathbf{A}^{-1}\|_1 = 1 + 1 + 2 + 2^2 + \ldots + 2^{n-2} = 2^{n-1}.$$

Таким образом, $\mathrm{cond}_1(\mathbf{A}) = n\,2^{n-1}$, т. е. матрица $\mathbf{A}$ плохо обусловлена.

Эти два примера показывают, что обусловленность матрицы слабо связана с величиной ее определителя.

Предположим, что в правую часть и матрицу системы (1.1) внесены возмущения $\Delta\mathbf{b}$ и $\Delta\mathbf{A}$ соответственно, которые вызвали изменение решения этой системы на величину $\Delta\mathbf{x}$. Таким образом, при условии, что $\mathbf{A} + \Delta\mathbf{A}$ – невырожденная матрица, имеем возмущенную систему

$$(\mathbf{A} + \Delta\mathbf{A})(\mathbf{x} + \Delta\mathbf{x}) = \mathbf{b} + \Delta\mathbf{b}.$$

Раскрывая скобки, получаем

$$\mathbf{A}\Delta\mathbf{x} = \Delta\mathbf{b} - \Delta\mathbf{A}\mathbf{x} - \Delta\mathbf{A}\Delta\mathbf{x} \quad \text{или} \quad \Delta\mathbf{x} = \mathbf{A}^{-1}\Delta\mathbf{b} - \mathbf{A}^{-1}\Delta\mathbf{A}\mathbf{x} - \mathbf{A}^{-1}\Delta\mathbf{A}\Delta\mathbf{x}.$$

Отсюда следует оценка

$$\|\Delta\mathbf{x}\| \le \|\mathbf{A}^{-1}\|\,\|\Delta\mathbf{b}\| + \|\mathbf{A}^{-1}\|\,\|\Delta\mathbf{A}\|\,\|\mathbf{x}\| + \|\mathbf{A}^{-1}\|\,\|\Delta\mathbf{A}\|\,\|\Delta\mathbf{x}\|$$

или

$$\begin{aligned}\big(1 - \|\mathbf{A}^{-1}\|\,\|\Delta\mathbf{A}\|\big)\frac{\|\Delta\mathbf{x}\|}{\|\mathbf{x}\|} &\le \|\mathbf{A}^{-1}\|\frac{\|\Delta\mathbf{b}\|}{\|\mathbf{x}\|} + \|\mathbf{A}^{-1}\|\,\|\Delta\mathbf{A}\| = \\ &= \|\mathbf{A}^{-1}\|\frac{\|\Delta\mathbf{b}\|}{\|\mathbf{b}\|}\frac{\|\mathbf{b}\|}{\|\mathbf{x}\|} + \|\mathbf{A}^{-1}\|\frac{\|\Delta\mathbf{A}\|}{\|\mathbf{A}\|}\|\mathbf{A}\|.\end{aligned}$$

Так как из (1.1) вытекает неравенство

$$\|\mathbf{b}\| \le \|\mathbf{A}\|\,\|\mathbf{x}\| \quad \text{или} \quad \frac{\|\mathbf{b}\|}{\|\mathbf{x}\|} \le \|\mathbf{A}\|,$$

то при условии $\|\mathbf{A}^{-1}\|\,\|\Delta\mathbf{A}\| < 1$ окончательно получаем

$$\frac{\|\Delta\mathbf{x}\|}{\|\mathbf{x}\|} \le \frac{\mathrm{cond}(\mathbf{A})}{1 - \mathrm{cond}(\mathbf{A})\|\Delta\mathbf{A}\|\,\|\mathbf{A}\|^{-1}}\Big(\frac{\|\Delta\mathbf{b}\|}{\|\mathbf{b}\|} + \frac{\|\Delta\mathbf{A}\|}{\|\mathbf{A}\|}\Big). \tag{5.4}$$

Таким образом, если матрица $\mathbf{A}$ хорошо обусловлена, то малое возмущение правой части и/или матрицы системы (1.1) может привести только к небольшому искажению решения. В противном случае эффект непредсказуем.

Пример 1.1. Рассмотрим решение линейной системы $\mathbf{A}\mathbf{x} = \mathbf{b}$, где

$$\mathbf{A} = \begin{pmatrix} 4{,}1 & 2{,}8 \\ 9{,}7 & 6{,}6 \end{pmatrix}, \quad \mathbf{b} = \begin{pmatrix} 4{,}1 \\ r9{,}7 \end{pmatrix}, \quad \mathbf{x} = \begin{pmatrix} 1 \\ 0 \end{pmatrix}.$$

Пользуясь первой нормой, имеем $\|\mathbf{b}\| = 13{,}8$ и $\|\mathbf{x}\| = 1$. Возьмем возмущенную правую часть

$$\tilde{\mathbf{b}} = \begin{pmatrix} 4{,}11 \\ 9{,}70 \end{pmatrix},$$

для которой простые вычисления дают новое решение

$$\tilde{\mathbf{x}} = \begin{pmatrix} 0{,}34 \\ 0{,}97 \end{pmatrix}.$$

Чтобы выяснить причину столь сильного изменения решения, воспользуемся оценкой (1.4).

Пусть $\Delta\mathbf{b} = \mathbf{b} - \tilde{\mathbf{b}}$ и $\Delta\mathbf{x} = \mathbf{x} - \tilde{\mathbf{x}}$. Тогда $\|\Delta\mathbf{b}\| = 0{,}01$, $\|\Delta\mathbf{x}\| = 1{,}63$ и

$$\frac{\|\Delta\mathbf{b}\|}{\|\mathbf{b}\|} = 0{,}0007246, \quad \frac{\|\Delta\mathbf{x}\|}{\|\mathbf{x}\|} = 1{,}63.$$

Поэтому согласно оценке (1.4)

$$\operatorname{cond}(\mathbf{A}) \geq \frac{1{,}63}{0{,}0007246} \approx 2249{,}4.$$

Следовательно, матрица нашей системы является плохообусловленной, что и объясняет столь сильное искажение решения при малом возмущении правой части системы.

Если приближенное решение $\tilde{\mathbf{x}}$ системы (1.1) найдено, то легко вычислить невязку

$$\mathbf{r} = \mathbf{b} - \mathbf{A}\tilde{\mathbf{x}} = \mathbf{A}(\mathbf{x}^* - \tilde{\mathbf{x}}) = -\mathbf{A}\mathbf{w}.$$

Отсюда для ошибки приближенного решения $\mathbf{w}$ получаем

$$\mathbf{w} = -\mathbf{A}^{-1}\mathbf{r} \quad \text{и} \quad \|\mathbf{w}\| \leq \|\mathbf{A}^{-1}\|\,\|\mathbf{r}\|.$$

Так как из системы (1.1) следует, что

$$\|\mathbf{b}\| \leq \|\mathbf{A}\|\,\|\mathbf{x}^*\|,$$

то, перемножая полученные неравенства, получаем

$$\|\mathbf{w}\|\,\|\mathbf{b}\| \leq \|\mathbf{A}^{-1}\|\,\|\mathbf{r}\|\,\|\mathbf{A}\|\,\|\mathbf{x}^*\|$$

или

$$\frac{\|\mathbf{w}\|}{\|\mathbf{x}^*\|} \leq \operatorname{cond}(\mathbf{A})\frac{\|\mathbf{r}\|}{\|\mathbf{b}\|}. \tag{5.5}$$

Таким образом, для хорошо обусловленной матрицы $\mathbf{A}$ относительная малость невязки влечет относительную малость ошибки приближенного решения. В противном случае результат непредсказуем.

Пример 1.2. Пусть для системы линейных уравнений

$$\begin{cases} 34x + 55y = 21, \\ 55x + 89y = 34. \end{cases}$$

имеются два приближенных решения

$$\tilde{\mathbf{x}}_1 = (-0{,}11; 0{,}45)^T \quad \text{и} \quad \tilde{\mathbf{x}}_2 = (-0{,}99; 1{,}01)^T.$$

Требуется выяснить какое из этих решений точнее, если проверка решений подстановкой дает невязки

$$\mathbf{r}_1 = \mathbf{b} - \mathbf{A}\tilde{\mathbf{x}}_1 = (-0{,}01; 0{,}00)^T \quad \text{и} \quad \mathbf{r}_2 = \mathbf{b} - \mathbf{A}\tilde{\mathbf{x}}_2 = (-0{,}89; -1{,}44)^T.$$

Пользуясь первой нормой, находим $\|\mathbf{r}_1\| = 0{,}01$, $\|\mathbf{r}_2\| = 2{,}33$ и казалось бы первое решение должно быть точнее.

Однако в данном случае точное решение $\mathbf{x}^* = (-1; 1)^T$. Для ошибок приближенных решений имеем:

$$\mathbf{w}_1 = \tilde{\mathbf{x}}_1 - \mathbf{x}^* = (0{,}89; -0{,}55)^T; \qquad \mathbf{w}_2 = \tilde{\mathbf{x}}_2 - \mathbf{x}^* = (0{,}01; 0{,}01)^T$$

и поэтому $\|\mathbf{w}_1\| = 1{,}44$, $\|\mathbf{w}_2\| = 0{,}02$. Таким образом, точнее оказывается второе решение.

Кажущееся противоречие легко объяснимо. В нашем примере матрица $\mathbf{A}$ плохо обусловлена: $\mathrm{cond}(\mathbf{A}) = 20736$ и поэтому согласно оценке (1.5) из малости невязки не обязательно следует малость ошибки приближенного решения.

Рассмотрим более подробно вопрос о малости определителя матрицы $\mathbf{A}$. Пусть $\mathbf{a}_i$ – столбцы матрицы $\mathbf{A}$. Если $\det(\mathbf{A}) \neq 0$, то векторы $\mathbf{a}_i$ линейно независимы и систему $\mathbf{Ax} = \mathbf{b}$ можно понимать как разложение вектора правой части $\mathbf{b}$ по вектор-столбцам матрицы $\mathbf{A}$:

$$\mathbf{b} = x_1\mathbf{a}_1 + x_2\mathbf{a}_2 + \ldots + x_n\mathbf{a}_n.$$

Определитель матрицы $\mathbf{A}$ – это объем n-мерного параллелепипеда, натянутого на векторы $\mathbf{a}_i$, $i = 1, 2, \ldots, n$. Этот объем может быть мал, если малы длины векторов $\mathbf{a}_i$ и/или углы между ними. От первого недостатка легко избавиться за счет нормировки векторов $\mathbf{a}_i$:

$$\mathbf{b} = y_1\overline{\mathbf{a}}_1 + y_2\overline{\mathbf{a}}_2 + \ldots + y_n\overline{\mathbf{a}}_n, \quad y_i = x_i\|\mathbf{a}_i\|, \quad \overline{\mathbf{a}}_i = \frac{\mathbf{a}_i}{\|\mathbf{a}_i\|}.$$

Малость определителя в новой системе:

$$\overline{\mathbf{A}}\mathbf{y} = \mathbf{b}, \quad \overline{\mathbf{A}} = (\overline{\mathbf{a}}_1, \overline{\mathbf{a}}_2, \ldots, \overline{\mathbf{a}}_n)$$

означает почти линейную зависимость векторов $\overline{\mathbf{a}}_i$. Как следствие, некоторые из компонент вектора $\mathbf{y}$ очень плохо представлены в математической модели, описываемой системой $\overline{\mathbf{A}}\mathbf{y} = \mathbf{b}$, которая является плохообусловленной. Чтобы избавиться от плохой обусловленности этой системы, можно внести изменения в используемую математическую модель или регуляризовать рассматриваемую систему линейных уравнений.

Основной вывод из рассмотрения, проведенного в этом параграфе, состоит в том, что непрерывную зависимость решения системы (1.1) от входных данных в общем случае можно обеспечить только для хорошо обусловленной матрицы $\mathbf{A}$, что достигается использованием «правильной» математической модели или применением метода регуляризации.

§ 5.4. Метод исключения Гаусса

Наиболее известным методом решения систем линейных уравнений вида (1.1) является метод исключения Гаусса. Перепишем систему (1.1) в покомпонентном виде

$$\begin{cases} a_{11}x_1 + a_{12}x_2 + \ldots + a_{1n}x_n = b_1, \\ a_{21}x_1 + a_{22}x_2 + \ldots + a_{2n}x_n = b_2, \\ \qquad\qquad \ldots \\ a_{n1}x_1 + a_{n2}x_2 + \ldots + a_{nn}x_n = b_n \end{cases} \tag{5.6}$$

или

$$\sum_{j=1}^{n} a_{ij}x_j = b_i, \quad i = 1, 2, \ldots, n.$$

Метод состоит в последовательном занулении коэффициентов системы (1.6), стоящих ниже главной диагонали матрицы этой системы, сначала в первом столбце, затем во втором столбце и т. д. с тем, чтобы привести эту систему к верхнему треугольному виду. Для этого используются линейные комбинации уравнений системы (1.6).

Пусть $a_{11} \neq 0$. Умножим первое уравнение системы (1.6) на величину $m_{i1} = a_{i1}/a_{11}$ и вычтем его из i-го уравнения для $i = 2, 3, \ldots, n$. В результате система (1.6) преобразуется к виду

$$\begin{cases} a_{11}x_1 + a_{12}x_2 + \ldots + a_{1n}x_n = b_1, \\ \qquad a_{22}^{(1)}x_2 + \ldots + a_{2n}^{(1)}x_n = b_2^{(1)}, \\ \qquad\qquad \ldots \\ \qquad a_{n2}^{(1)}x_2 + \ldots + a_{nn}^{(1)}x_n = b_n^{(1)}, \end{cases} \tag{5.7}$$

где

$$a_{ij}^{(1)} = a_{ij} - m_{i1}a_{1j}; \quad b_i^{(1)} = b_i - m_{i1}b_1; \quad i,j = 2,3,\ldots,n.$$

Пусть $a_{22}^{(1)} \neq 0$. Умножая второе уравнение системы (1.7) на $m_{i2} = a_{i2}^{(1)}/a_{22}^{(1)}$ и вычитая его из i-го уравнения для $i = 3,4,\ldots,n$, занулим коэффициенты второго столбца системы (1.7), стоящие ниже элемента $a_{22}^{(1)}$. При этом преобразуемые коэффициенты и правая часть вычисляются по формулам

$$a_{ij}^{(2)} = a_{ij}^{(1)} - m_{i2}a_{2j}^{(1)}, \quad b_i^{(2)} = b_i^{(1)} - m_{i2}b_2^{(1)}, \quad i,j = 3,4,\ldots,n.$$

Продолжая этот процесс, приходим к системе с верхней треугольной матрицей, эквивалентной системе (1.6)

$$\begin{cases} a_{11}x_1 + a_{12}x_2 + \ldots + a_{1n}x_n = b_1, \\ \qquad a_{22}^{(1)}x_2 + \ldots + a_{2n}^{(1)}x_n = b_2^{(1)}, \\ \qquad\qquad \ldots \\ \qquad\qquad\quad a_{nn}^{(n-1)}x_n = b_n^{(n-1)}, \end{cases} \tag{5.8}$$

где

$$\begin{aligned} a_{ij}^{(i-1)} &= a_{ij}^{(i-2)} - m_{i,i-1}a_{i-1,j}^{(i-2)}, \quad j = i, i+1, \ldots, n; \\ b_i^{(i-1)} &= b_i^{(i-2)} - m_{i,i-1}b_{i-1}^{(i-2)}, \quad i = 3,4,\ldots,n. \end{aligned}$$

Стоящие на диагонали коэффициенты этой системы $a_{11}, a_{22}^{(1)}, \ldots, a_{nn}^{(n-1)}$, которые по предположению все отличны от нуля, принято называть *ведущими элементами* метода исключения Гаусса. Их произведение дает нам определитель матрицы системы (1.8), который в силу использованных преобразований и свойств определителей совпадает с определителем системы (1.6). Поэтому

$$\det(\mathbf{A}) = a_{11}a_{22}^{(1)}\ldots a_{nn}^{(n-1)}.$$

Решение системы (1.8) может быть получено по рекуррентным формулам:

$$\begin{aligned} x_n &= b_n^{(n-1)}/a_{nn}^{(n-1)}; \\ x_i &= \frac{1}{a_{ii}^{(i-1)}}\Big[b_i^{(i-1)} - \sum_{j=i+1}^{n} a_{ij}^{(i-1)}x_j\Big], \quad i = n-1, n-2, \ldots, 1. \end{aligned} \tag{5.9}$$

Приведение системы (1.6) к треугольному виду (1.8) принято называть *прямым ходом* метода исключения Гаусса. Решение системы (1.8) по рекуррентным формулам (1.9) называют *обратным ходом* этого метода.

Вычисления по методу Гаусса обычно проводят, используя расширенную матрицу

$$\overline{\mathbf{A}} = (\mathbf{A} \mid \mathbf{b}) = \left(\begin{array}{cccc|c} a_{11} & a_{12} & \dots & a_{1n} & b_1 \\ a_{21} & a_{22} & \dots & a_{2n} & b_2 \\ & & \dots & & \dots \\ a_{n1} & a_{n2} & \dots & a_{nn} & b_n \end{array}\right) \quad \to \quad (\mathbf{E} \mid \mathbf{x}).$$

Проводя над строками матрицы $\overline{\mathbf{A}}$ линейные операции метода исключения Гаусса, матрицу $\mathbf{A}$ вначале преобразуем к верхней треугольной матрице а затем с помощью обратного хода к единичной матрице $\mathbf{E}$. При этом на месте столбца правой части $\mathbf{b}$ получим решение $\mathbf{x}$ системы (1.6).

Метод исключения Гаусса может быть также применен для нахождения обратной матрицы $\mathbf{A}^{-1}$ системы (1.6). Для этого используется матричное равенство

$$\mathbf{AX} = \mathbf{E}$$

или в покомпонентной записи

$$\begin{pmatrix} a_{11} & a_{12} & \dots & a_{1n} \\ a_{21} & a_{22} & \dots & a_{2n} \\ & & \dots & \\ a_{n1} & a_{n2} & \dots & a_{nn} \end{pmatrix} \begin{pmatrix} x_{11} & x_{12} & \dots & x_{1n} \\ x_{21} & x_{22} & \dots & x_{2n} \\ & & \dots & \\ x_{n1} & x_{n2} & \dots & x_{nn} \end{pmatrix} = \begin{pmatrix} 1 & 0 & \dots & 0 \\ 0 & 1 & \dots & 0 \\ & & \dots & \\ 0 & 0 & \dots & 1 \end{pmatrix},$$

которое равносильно n системам вида $\mathbf{Ax}_j = \mathbf{e}_j$ с вектор-столбцами неизвестных $\mathbf{x}_j = (x_{1j}, x_{2j}, \dots, x_{nj})^T$ и правых частей $\mathbf{e}_j = (0, \dots, \underset{j}{1}, \dots, 0)^T$. Поэтому для нахождения обратной матрицы $\mathbf{A}^{-1}$ достаточно применить метод исключения Гаусса к расширенной матрице

$$(\mathbf{A} \mid \mathbf{E}) = \left(\begin{array}{cccc|cccc} a_{11} & a_{12} & \dots & a_{1n} & 1 & 0 & \dots & 0 \\ a_{21} & a_{22} & \dots & a_{2n} & 0 & 1 & \dots & 0 \\ & & \dots & & & & \dots & \\ a_{n1} & a_{n2} & \dots & a_{nn} & 0 & 0 & \dots & 1 \end{array}\right) \quad \to \quad (\mathbf{E} \mid \mathbf{A}^{-1}).$$

Проводя исключение, на месте матрицы $\mathbf{A}$ получим единичную матрицу $\mathbf{E}$, а единичная матрица преобразуется к обратной матрице $\mathbf{A}^{-1}$.

Нетрудно подсчитать число арифметических операций N, необходимых для решения системы вида (1.6) методом исключения Гаусса. При прямом ходе проводится $(n-1)n/2$ делений

$$(n-1)^2 + (n-2)^2 + \ldots + 1^2 + \frac{(n-1)n}{2} = \frac{(n-1)n(2n-1)}{6} + \frac{(n-1)n}{2},$$

умножений и столько же вычитаний. При обратном ходе выполняется n делений, $(n-1)n/2$ умножений и столько же вычитаний. Таким образом, общее число арифметических операций в методе Гаусса

$$N = \frac{(n-1)n(2n-1)}{3} + \frac{5}{2}(n-1)n + n < n^3 \quad \text{при} \quad n \geq 4.$$

Если требуется решить m систем вида (1.6) с одной и той же матрицей и разными правыми частями, то в этом случае

$$N = \frac{(n-1)n(2n-1)}{3} + \frac{(n-1)n}{2} + mn(2n-1).$$

§ 5.5. Матричная формулировка гауссова исключения

Рассмотренный выше процесс приведения матрицы $\mathbf{A}$ системы (1.6) к верхней треугольной форме в действительности эквивалентен последовательному умножению этой матрицы на нижние треугольные матрицы

$$\mathbf{L}_{n-1}\mathbf{L}_{n-2}\dots\mathbf{L}_1\mathbf{A} = \mathbf{U}, \tag{5.10}$$

где матрица $\mathbf{L}_j$, $j = 1, 2, \dots, n-1$ отличается от единичной матрицы наличием поддиагональных элементов в j-м столбце

$$\mathbf{L}_j = \begin{pmatrix} 1 & & & & \\ & \ddots & & 0 & \\ & & 1 & & \\ & & -m_{j+1,j} & & \\ & & \vdots & \ddots & \\ & 0 & -m_{n,j} & 0 & 1 \end{pmatrix}, \quad m_{i,j} = \frac{a_{ij}^{(j-1)}}{a_{jj}^{(j-1)}}, \quad i = j+1, j+2, \dots, n.$$

Матрицы $\mathbf{L}_j$ обладают двумя замечательными свойствами:

1) обратная матрица $\mathbf{L}_j^{-1}$ получается из $\mathbf{L}_j$ заменой знаков поддиагональных элементов в j-м столбце;

2) произведение матриц $\mathbf{L}_j^{-1}\mathbf{L}_{j+1}^{-1}$ дает нижнюю треугольную матрицу, которая отличается от единичной матрицы наличием поддиагональных элементов матриц $\mathbf{L}_j^{-1}$ и $\mathbf{L}_{j+1}^{-1}$ на их обычных местах.

Нетрудно понять почему это происходит. Пусть $\mathbf{e}_j$ – j-й столбец единичной матрицы $\mathbf{E}$ и $\mathbf{l}_j = (0, \dots, 0, m_{j+1,j}, \dots, m_{n,j})^T$. Тогда $\mathbf{L}_j = \mathbf{E} - \mathbf{l}_j\mathbf{e}_j^T$. Поскольку $\mathbf{e}_j^T\mathbf{l}_j = 0$, то

$$(\mathbf{E} - \mathbf{l}_j\mathbf{e}_j^T)(\mathbf{E} + \mathbf{l}_j\mathbf{e}_j^T) = \mathbf{E} - \mathbf{l}_j\mathbf{e}_j^T\mathbf{l}_j\mathbf{e}_j^T = \mathbf{E},$$

т. е. обратная матрица $\mathbf{L}_j^{-1} = \mathbf{E} + \mathbf{l}_j\mathbf{e}_j^T$.

Для произведения матриц $\mathbf{L}_j^{-1}$ и $\mathbf{L}_{j+1}^{-1}$ получаем

$$\mathbf{L}_j^{-1}\mathbf{L}_{j+1}^{-1} = (\mathbf{E} + \mathbf{l}_j\mathbf{e}_j^T)(\mathbf{E} + \mathbf{l}_{j+1}\mathbf{e}_{j+1}^T) = \mathbf{E} + \mathbf{l}_j\mathbf{e}_j^T + \mathbf{l}_{j+1}\mathbf{e}_{j+1}^T.$$

Теперь, полагая $\mathbf{L} = \mathbf{L}_1^{-1}\mathbf{L}_2^{-1}\dots\mathbf{L}_{n-1}^{-1}$, из (1.10) получаем

$$\mathbf{A} = \mathbf{L}\mathbf{U},$$

где

$$\mathbf{L} = \begin{pmatrix} 1 & 0 & \dots & 0 \\ m_{2,1} & 1 & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ m_{n,1} & m_{n,2} & \dots & 1 \end{pmatrix}, \quad \mathbf{U} = \begin{pmatrix} a_{11} & a_{12} & \dots & a_{1n} \\ 0 & a_{22}^{(1)} & \dots & a_{2n}^{(1)} \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & a_{nn}^{(n-1)} \end{pmatrix}.$$

Это так называемое **LU**-разложение матрицы **A** на нижнюю и верхнюю треугольные матрицы **L** и **U** соответственно. Записывая теперь систему (1.6) в виде

$$\mathbf{LUx} = \mathbf{b}$$

и полагая $\mathbf{Ux} = \mathbf{y}$, мы сводим решение этой системы к решению двух систем с треугольными матрицами

$$\mathbf{Ly} = \mathbf{b} \quad \text{и} \quad \mathbf{Ux} = \mathbf{y}.$$

Решение этих систем называют *прямой и обратной подстановками* метода исключения Гаусса. Оно может быть легко получено за n^2 арифметических операций.

При компьютерных вычислениях элементы m_{ij} матрицы **L** хранятся на месте поддиагональных элементов матрицы **A**, а элементы матрицы **U** располагаются на месте диагональных и наддиагональных элементов **A**. В результате на месте **A** получаем матрицу

$$\mathbf{L} - \mathbf{E} + \mathbf{U} = \begin{pmatrix} a_{11} & a_{12} & \dots & a_{1n} \\ m_{21} & a_{22}^{(1)} & \dots & a_{2n}^{(1)} \\ \vdots & \vdots & \ddots & \vdots \\ m_{n1} & m_{n2} & \dots & a_{nn}^{(n-1)} \end{pmatrix}.$$

Это позволяет существенно экономить память ЭВМ.

Пример 1.3. Рассмотрим получение LU-разложения для матрицы

$$\mathbf{A} = \begin{pmatrix} 2 & 1 & 1 & 0 \\ 4 & 3 & 3 & 1 \\ 8 & 7 & 9 & 5 \\ 6 & 7 & 9 & 8 \end{pmatrix}. \tag{5.11}$$

На первом шаге гауссова исключения имеем

$$\mathbf{L}_1\mathbf{A} = \begin{pmatrix} 1 & 0 & 0 & 0 \\ -2 & 1 & 0 & 0 \\ -4 & 0 & 1 & 0 \\ -3 & 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} 2 & 1 & 1 & 0 \\ 4 & 3 & 3 & 1 \\ 8 & 7 & 9 & 5 \\ 6 & 7 & 9 & 8 \end{pmatrix} = \begin{pmatrix} 2 & 1 & 1 & 0 \\ 0 & 1 & 1 & 1 \\ 0 & 3 & 5 & 5 \\ 0 & 4 & 6 & 8 \end{pmatrix}.$$

Второй шаг гауссова исключения выглядит следующим образом

$$\mathbf{L}_2\mathbf{L}_1\mathbf{A} = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & -3 & 1 & 0 \\ 0 & -4 & 0 & 1 \end{pmatrix} \begin{pmatrix} 2 & 1 & 1 & 0 \\ 0 & 1 & 1 & 1 \\ 0 & 3 & 5 & 5 \\ 0 & 4 & 6 & 8 \end{pmatrix} = \begin{pmatrix} 2 & 1 & 1 & 0 \\ 0 & 1 & 1 & 1 \\ 0 & 0 & 2 & 2 \\ 0 & 0 & 2 & 4 \end{pmatrix}.$$

Наконец, на третьем заключительном шаге гауссова исключения получаем

$$\mathbf{L}_3\mathbf{L}_2\mathbf{L}_1\mathbf{A} = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & -1 & 1 \end{pmatrix} \begin{pmatrix} 2 & 1 & 1 & 0 \\ 0 & 1 & 1 & 1 \\ 0 & 0 & 2 & 2 \\ 0 & 0 & 2 & 4 \end{pmatrix} = \begin{pmatrix} 2 & 1 & 1 & 0 \\ 0 & 1 & 1 & 1 \\ 0 & 0 & 2 & 2 \\ 0 & 0 & 0 & 2 \end{pmatrix} = \mathbf{U}.$$

Теперь для получения разложения $\mathbf{A} = \mathbf{LU}$ нам остается вычислить матрицу $\mathbf{L} = \mathbf{L}_1^{-1}\mathbf{L}_2^{-1}\mathbf{L}_3^{-1}$. Согласно сказанному выше, обратная матрица $\mathbf{L}_1^{-1}$ получается из $\mathbf{L}_1$ заменой в последней знаков поддиагональных элементов:

$$\mathbf{L}_1^{-1} = \begin{pmatrix} 1 & 0 & 0 & 0 \\ -2 & 1 & 0 & 0 \\ -4 & 0 & 1 & 0 \\ -3 & 0 & 0 & 1 \end{pmatrix}^{-1} = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 2 & 1 & 0 & 0 \\ 4 & 0 & 1 & 0 \\ 3 & 0 & 0 & 1 \end{pmatrix}.$$

Аналогично получаем матрицы $\mathbf{L}_2^{-1}$ и $\mathbf{L}_3^{-1}$. Наконец, нижняя треугольная матрица $\mathbf{L} = \mathbf{L}_1^{-1}\mathbf{L}_2^{-1}\mathbf{L}_3^{-1}$ получается из единичной матрицы заменой поддиагональных элементов на соответствующие ненулевые элементы матриц $\mathbf{L}_1^{-1}$, $\mathbf{L}_2^{-1}$ и $\mathbf{L}_3^{-1}$ на их обычных местах

$$\mathbf{A} = \begin{pmatrix} 2 & 1 & 1 & 0 \\ 4 & 3 & 3 & 1 \\ 8 & 7 & 9 & 5 \\ 6 & 7 & 9 & 8 \end{pmatrix} = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 2 & 1 & 0 & 0 \\ 4 & 3 & 1 & 0 \\ 3 & 4 & 1 & 1 \end{pmatrix} \begin{pmatrix} 2 & 1 & 1 & 0 \\ 0 & 1 & 1 & 1 \\ 0 & 0 & 2 & 2 \\ 0 & 0 & 0 & 2 \end{pmatrix} = \mathbf{LU}. \tag{5.12}$$

В компактной форме матрица LU-разложения принимает вид

$$\mathbf{L} - \mathbf{E} + \mathbf{U} = \begin{pmatrix} 2 & 1 & 1 & 0 \\ 2 & 1 & 1 & 1 \\ 4 & 3 & 2 & 2 \\ 3 & 4 & 1 & 2 \end{pmatrix}.$$

§ 5.6. Исключение с выбором ведущего элемента

Рассмотренный нами алгоритм гауссова исключения можно применить только в предположении, что все ведущие элементы $a_{11}, a_{22}^{(1)}, \ldots, a_{nn}^{(n-1)}$ отличны от нулю. Напомним, что матрица $\mathbf{A}$ считается невырожденной, а $\det(\mathbf{A}) = a_{11}a_{22}^{(1)}, \ldots, a_{nn}^{(n-1)}$.

Пусть теперь $a_{11} = 0$. Так как $\det(\mathbf{A}) \neq 0$, то для некоторого $i > 1$ найдется элемент $a_{i1} \neq 0$. Если поменять местами первую и i-ю строки в $[\mathbf{A}, \mathbf{b}]$, то получим эквивалентную систему уравнений с $a_{11} \neq 0$. Для этой системы уже применим описанный выше алгоритм исключения. Аналогичный прием можно повторить на любом шаге, когда $a_{jj}^{(j-1)} = 0$. К сожалению, использование близких к нулю ведущих элементов может привести к неприятностям.

Пример 1.4. Рассмотрим систему уравнений

$$\begin{pmatrix} 0 & 1 \\ 1 & 1 \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \end{pmatrix} = \begin{pmatrix} 1 \\ 2 \end{pmatrix},$$

где матрица невырождена и хорошо обусловлена:

$$\mathrm{cond}_2(\mathbf{A}) = (3 + \sqrt{5})/2 \approx 2,618.$$

Здесь $a_{11} = 0$ и исключение не может быть проведено, так как оно связано с делением на нуль. Переставив однако местами уравнения, получаем точное решение $\mathbf{x}^* = (1; 1)^T$.

Рассмотрим теперь возмущенную систему

$$\begin{pmatrix} \varepsilon & 1 \\ 1 & 1 \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \end{pmatrix} = \begin{pmatrix} 1 \\ 2 \end{pmatrix}, \quad \varepsilon = 10^{-10}.$$

Проводя исключение, получаем

$$\begin{pmatrix} \varepsilon & 1 \\ 0 & 1 - \varepsilon^{-1} \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \end{pmatrix} = \begin{pmatrix} 1 \\ 2 - \varepsilon^{-1} \end{pmatrix}.$$

Если теперь вычисления проводятся, например, с девятью значащими десятичными цифрами, то числа $1 - \varepsilon^{-1}$ и $2 - \varepsilon^{-1}$ будут округлены до ближайших целых. Пусть это будет $-\varepsilon^{-1}$. Тогда мы получим приближенное решение $\mathbf{x} = (0; 1)^T$, что совершенно неудовлетворительно, так как точное решение нашей системы $\mathbf{x}^* = (1; 1)^T$.

Чтобы избежать этой неприятности, переставим опять уравнения в нашей системе и, проводя исключение, получим

$$\begin{pmatrix} 1 & 1 \\ 0 & 1 - \varepsilon \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \end{pmatrix} = \begin{pmatrix} 2 \\ 1 - 2\varepsilon \end{pmatrix}.$$

Теперь находим правильное решение $\mathbf{x} = (1; 1)^T$.

Из приведенного примера следует, что нужно избегать малых по абсолютной величине ведущих элементов $a_{jj}^{(j-1)}$. В качестве ведущего элемента рекомендуется выбирать максимальное по абсолютной величине число в j-м столбце, т. е.

$a_{jj}^{(j-1)} = \max_i |a_{ij}^{(j-1)}|$. Это достигается на каждом шаге исключения перестановкой j-й и i-й строк. Такая стратегия частичного упорядочивания называется *методом исключения с выбором ведущего элемента по столбцам.* Может быть рассмотрена и стратегия выбора ведущего элемента по строкам, когда $a_{ii}^{(i-1)} = \max_j |a_{ij}^{(i-1)}|$. Гораздо реже используется стратегия полного упорядочивания, при которой в качестве ведущего элемента берется $a_{jj}^{(j-1)} = \max_{i,j} |a_{ij}^{(j-1)}|$. Этот подход называется *методом исключения с полным выбором ведущего элемента.* После выбора ведущего элемента зануление элементов j-го столбца осуществляется по стандартной схеме исключения.

Перестановка строк (столбцов) матрицы $\mathbf{A}$ порядка $n \times n$ может быть осуществлена путем умножения этой матрицы слева (справа) на матрицу $\mathbf{P}$, которая получается из единичной матрицы порядка $n \times n$ перестановкой тех же строк (столбцов). В общем случае *матрицей перестановок* $\mathbf{P}$ порядка $n \times n$ называют такую матрицу, в которой каждая строка и каждый столбец содержат одну единицу и $n - 1$ нулей.

На практике наиболее распространен метод исключения с выбором ведущего элемента по столбцам. В этом случае на j-м шаге вначале осуществляется выбор ведущего элемента путем умножения слева на матрицу перестановок $\mathbf{P}_j$ а затем проводится исключение путем умножения слева на нижнюю треугольную матрицу $\mathbf{L}_j$. После $n - 1$ шагов матрица $\mathbf{A}$ переходит в верхнюю треугольную матрицу $\mathbf{U}$:

$$\mathbf{L}_{n-1}\mathbf{P_{n-1}} \ldots \mathbf{L}_2\mathbf{P_2}\mathbf{L}_1\mathbf{P_1}\mathbf{A} = \mathbf{U}. \tag{5.13}$$

Наличие здесь матриц перестановок $\mathbf{P}_j$ осложняет процесс исключения. Вообще говоря, мы уже не получаем LU-разложение матрицы $\mathbf{A}$ как произведение нижней и верхней треугольных матриц. Такое разложение однако можно получить за счет соответствующей перестановки строк и столбцов матрицы $\mathbf{A}$ до начала процесса исключения. К сожалению, это преобразование матрицы $\mathbf{A}$ не всегда единственно и к тому же трудно осуществимо из-за сложности процесса исключения. Приведем следующий результат, конструктивное доказательство которого будет дано ниже.

Теорема 1.1. *Если* $\mathbf{A}$ *– невырожденная квадратная матрица, то существует матрица перестановок* $\mathbf{P}$*, невырожденная нижняя треугольная матрица* $\mathbf{L}$ *с единичной диагональю и невырожденная верхняя треугольная матрица* $\mathbf{U}$ *такие, что*

$$\mathbf{PA} = \mathbf{LU}.$$

Пример 1.5. Рассмотрим опять матрицу (1.11)

$$\mathbf{A} = \begin{pmatrix} 2 & 1 & 1 & 0 \\ 4 & 3 & 3 & 1 \\ 8 & 7 & 9 & 5 \\ 6 & 7 & 9 & 8 \end{pmatrix},$$

которую преобразуем к верхней треугольной форме, используя метод исключения с выбором ведущего элемента по столбцам.

Вначале выберем в качестве ведущего элемента наибольшее из чисел в первом столбце, переставив для этого местами первую и третью строки путем умножения слева на матрицу перестановок $\mathbf{P}_1$:

$$\mathbf{P}_1\mathbf{A} = \begin{pmatrix} 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 \\ 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} 2 & 1 & 1 & 0 \\ 4 & 3 & 3 & 1 \\ 8 & 7 & 9 & 5 \\ 6 & 7 & 9 & 8 \end{pmatrix} = \begin{pmatrix} 8 & 7 & 9 & 5 \\ 4 & 3 & 3 & 1 \\ 2 & 1 & 1 & 0 \\ 6 & 7 & 9 & 8 \end{pmatrix}.$$

Исключение в первом столбце проведем с помощью умножения слева на нижнюю треугольную матрицу $\mathbf{L}_1$:

$$\mathbf{L}_1\mathbf{P}_1\mathbf{A} = \begin{pmatrix} 1 & 0 & 0 & 0 \\ -\frac{1}{2} & 1 & 0 & 0 \\ -\frac{1}{4} & 0 & 1 & 0 \\ -\frac{3}{4} & 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} 8 & 7 & 9 & 5 \\ 4 & 3 & 3 & 1 \\ 2 & 1 & 1 & 0 \\ 6 & 7 & 9 & 8 \end{pmatrix} = \begin{pmatrix} 8 & 7 & 9 & 5 \\ 0 & -\frac{1}{2} & -\frac{3}{2} & -\frac{3}{2} \\ 0 & -\frac{3}{4} & -\frac{5}{4} & -\frac{5}{4} \\ 0 & \frac{7}{4} & \frac{9}{4} & \frac{17}{4} \end{pmatrix}.$$

Теперь вторая и четвертая строки в полученной матрице переставляются с помощью умножения на матрицу $\mathbf{P}_2$:

$$\mathbf{P}_2\mathbf{L}_1\mathbf{P}_1\mathbf{A} = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 \end{pmatrix} \begin{pmatrix} 8 & 7 & 9 & 5 \\ 0 & -\frac{1}{2} & -\frac{3}{2} & -\frac{3}{2} \\ 0 & -\frac{3}{4} & -\frac{5}{4} & -\frac{5}{4} \\ 0 & \frac{7}{4} & \frac{9}{4} & \frac{17}{4} \end{pmatrix} = \begin{pmatrix} 8 & 7 & 9 & 5 \\ 0 & \frac{7}{4} & \frac{9}{4} & \frac{17}{4} \\ 0 & -\frac{3}{4} & -\frac{5}{4} & -\frac{5}{4} \\ 0 & -\frac{1}{2} & -\frac{3}{2} & -\frac{3}{2} \end{pmatrix}.$$

Второй шаг исключения проводим путем умножения на матрицу $\mathbf{L}_2$:

$$\mathbf{L}_2\mathbf{P}_2\mathbf{L}_1\mathbf{P}_1\mathbf{A} = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & \frac{3}{7} & 1 & 0 \\ 0 & \frac{2}{7} & 0 & 1 \end{pmatrix} \begin{pmatrix} 8 & 7 & 9 & 5 \\ 0 & \frac{7}{4} & \frac{9}{4} & \frac{17}{4} \\ 0 & -\frac{3}{4} & -\frac{5}{4} & -\frac{5}{4} \\ 0 & -\frac{1}{2} & -\frac{3}{2} & -\frac{3}{2} \end{pmatrix} = \begin{pmatrix} 8 & 7 & 9 & 5 \\ 0 & \frac{7}{4} & \frac{9}{4} & \frac{17}{4} \\ 0 & 0 & -\frac{2}{7} & \frac{4}{7} \\ 0 & 0 & -\frac{6}{7} & -\frac{2}{7} \end{pmatrix}.$$

Третья и четвертая строки переставляются умножением на матрицу $\mathbf{P}_3$:

$$\mathbf{P}_3\mathbf{L}_2\mathbf{P}_2\mathbf{L}_1\mathbf{P}_1\mathbf{A} = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{pmatrix} \begin{pmatrix} 8 & 7 & 9 & 5 \\ 0 & \frac{7}{4} & \frac{9}{4} & \frac{17}{4} \\ 0 & 0 & -\frac{2}{7} & \frac{4}{7} \\ 0 & 0 & -\frac{6}{7} & -\frac{2}{7} \end{pmatrix} = \begin{pmatrix} 8 & 7 & 9 & 5 \\ 0 & \frac{7}{4} & \frac{9}{4} & \frac{17}{4} \\ 0 & 0 & -\frac{6}{7} & -\frac{2}{7} \\ 0 & 0 & -\frac{2}{7} & \frac{4}{7} \end{pmatrix}.$$

На заключительном шаге исключения умножаем на матрицу $\mathbf{L}_3$:

$$\mathbf{L}_3\mathbf{P}_3\mathbf{L}_2\mathbf{P}_2\mathbf{L}_1\mathbf{P}_1\mathbf{A} = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & -\frac{1}{3} & 1 \end{pmatrix} \begin{pmatrix} 8 & 7 & 9 & 5 \\ 0 & \frac{7}{4} & \frac{9}{4} & \frac{17}{4} \\ 0 & 0 & -\frac{6}{7} & -\frac{2}{7} \\ 0 & 0 & -\frac{2}{7} & \frac{4}{7} \end{pmatrix} = \begin{pmatrix} 8 & 7 & 9 & 5 \\ 0 & \frac{7}{4} & \frac{9}{4} & \frac{17}{4} \\ 0 & 0 & -\frac{6}{7} & -\frac{2}{7} \\ 0 & 0 & 0 & \frac{2}{3} \end{pmatrix}.$$

Таким образом, мы получили равенство

$$\mathbf{L}_3\mathbf{P}_3\mathbf{L}_2\mathbf{P}_2\mathbf{L}_1\mathbf{P}_1\mathbf{A} = \mathbf{U}, \tag{5.14}$$

которое, вообще говоря, не дает LU-разложения матрицы $\mathbf{A}$.

Покажем, что равенство (1.14) можно переписать в виде LU-разложения матрицы $\mathbf{A}$:

$$\begin{pmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 1 & 0 & 0 \\ 1 & 0 & 0 & 0 \end{pmatrix} \begin{pmatrix} 2 & 1 & 1 & 0 \\ 4 & 3 & 3 & 1 \\ 8 & 7 & 9 & 5 \\ 6 & 7 & 9 & 8 \end{pmatrix} = \begin{pmatrix} 1 & 0 & 0 & 0 \\ \frac{3}{4} & 1 & 0 & 0 \\ \frac{1}{2} & -\frac{2}{7} & 1 & 0 \\ \frac{1}{4} & -\frac{3}{7} & \frac{1}{3} & 1 \end{pmatrix} \begin{pmatrix} 8 & 7 & 9 & 5 \\ 0 & \frac{7}{4} & \frac{9}{4} & \frac{17}{4} \\ 0 & 0 & -\frac{6}{7} & -\frac{2}{7} \\ 0 & 0 & 0 & \frac{2}{3} \end{pmatrix} \tag{5.15}$$

или

$$\mathbf{PA} = \mathbf{LU}. \tag{5.16}$$

Отметим вначале особенности разложения (1.15). В этом представлении все поддиагональные элементы матрицы $\mathbf{L}$ по абсолютной величине не превосходят единицу. Это следствие того факта, что в методе исключения с выбором ведущих элементов по строкам в качестве последних выбираются наибольшие из чисел в соответствующих столбцах $a_{jj}^{(j-1)} = \max_i |a_{ij}^{(j-1)}|$.

Покажем теперь как получить равенство (1.16) из (1.14). Шесть элементарных матриц в (1.14) можно переписать в виде

$$\mathbf{L}_3\mathbf{P}_3\mathbf{L}_2\mathbf{P}_2\mathbf{L}_1\mathbf{P}_1 = \mathbf{L}_3'\mathbf{L}_2'\mathbf{L}_1'\mathbf{P}_3\mathbf{P}_2\mathbf{P}_1,$$

где

$$\mathbf{L}_3' = \mathbf{L}_3, \quad \mathbf{L}_2' = \mathbf{P}_3\mathbf{L}_2\mathbf{P}_3^{-1}, \quad \mathbf{L}_1' = \mathbf{P}_3\mathbf{P}_2\mathbf{L}_1\mathbf{P}_2^{-1}\mathbf{P}_3^{-1}.$$

В этих равенствах применение матриц перестановок $\mathbf{P}_i$ при $i > j$ не меняет структуру матриц $\mathbf{L}_j$:

$$\mathbf{L}'_2 = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & \frac{2}{7} & 1 & 0 \\ 0 & \frac{3}{7} & 0 & 1 \end{pmatrix}, \quad \mathbf{L}'_1 = \begin{pmatrix} 1 & 0 & 0 & 0 \\ -\frac{3}{4} & 1 & 0 & 0 \\ -\frac{1}{2} & 0 & 1 & 0 \\ -\frac{1}{4} & 0 & 0 & 1 \end{pmatrix}.$$

Поэтому матрицы $\mathbf{L}'_j$ имеют ту же структуру, что и матрицы $\mathbf{L}_j$. Вычисление произведения матриц $\mathbf{L}'_j$ и $\mathbf{P}_j$ дает

$$\begin{aligned} \mathbf{L}'_3\mathbf{L}'_2\mathbf{L}'_1\mathbf{P}_3\mathbf{P}_2\mathbf{P}_1 &= \mathbf{L}_3(\mathbf{P}_3\mathbf{L}_2\mathbf{P}_3^{-1})(\mathbf{P}_3\mathbf{P}_2\mathbf{L}_1\mathbf{P}_2^{-1}\mathbf{P}_3^{-1})\mathbf{P}_3\mathbf{P}_2\mathbf{P}_1 = \\ &= \mathbf{L}_3\mathbf{P}_3\mathbf{L}_2\mathbf{P}_2\mathbf{L}_1\mathbf{P}_1. \end{aligned}$$

Таким образом, в равенстве (1.16) имеем $\mathbf{P} = \mathbf{P}_3\mathbf{P}_2\mathbf{P}_1$ и $\mathbf{L} = (\mathbf{L}'_3\mathbf{L}'_2\mathbf{L}'_1)^{-1}$.

В случае матрицы $\mathbf{A}$ порядка $n \times n$ разложение (1.13), получаемое методом исключения с выбором ведущих элементов по столбцам, может быть записано в виде

$$(\mathbf{L}'_{n-1} \dots \mathbf{L}'_2\mathbf{L}'_1)(\mathbf{P}_{n-1} \dots \mathbf{P}_1\mathbf{P}_1)\mathbf{A} = \mathbf{U}, \tag{5.17}$$

где матрица $\mathbf{L}'_j$ определяется равенством

$$\mathbf{L}'_j = \mathbf{P}_{n-1} \dots \mathbf{P}_{j+1}\mathbf{L}_j\mathbf{P}_{j+1}^{-1} \dots \mathbf{P}_{n-1}^{-1}. \tag{5.18}$$

Произведение матриц $\mathbf{L}'_j$ дает нижнюю треугольную матрицу с единичной диагональю, которая может быть легко обращена путем изменения знаков поддиагональных элементов как это делалось выше в гауссовом исключении без выбора ведущих элементов. Полагая теперь $\mathbf{L} = (\mathbf{L}'_{n-1} \dots \mathbf{L}'_2\mathbf{L}'_1)^{-1}$ и $\mathbf{P} = (\mathbf{P}_{n-1} \dots \mathbf{P}_2\mathbf{P}_1)^{-1}$, получаем

$$\mathbf{PA} = \mathbf{LU}. \tag{5.19}$$

Приведенное построение доказывает справедливость теоремы 1.1.

В общем случае всякая квадратная матрица независимо от того вырождена она или нет может быть представлена в виде (1.19), где $\mathbf{P}$ – матрица перестановок, $\mathbf{L}$ – нижняя треугольная матрица с единичной диагональю, поддиагональные элементы которой по абсолютной величине не больше единицы, и $\mathbf{U}$ – верхняя треугольная матрица. Формулу (1.19) принято называть LU-разложением матрицы $\mathbf{A}$. Для экономии компьютерной памяти матрицы $\mathbf{L}$ и $\mathbf{U}$ опять могут быть помещены на место матрицы $\mathbf{A}$.

В случае редко применяемого на практике полного выбора ведущих элементов каждому шагу исключения предшествует умножение на матрицу перестановок строк $\mathbf{P}_j$ слева и столбцов $\mathbf{Q}_j$ справа. В результате получаем

$$\mathbf{L}_{n-1}\mathbf{P}_{n-1} \dots \mathbf{L}_2\mathbf{P}_2\mathbf{L}_1\mathbf{P}_1\mathbf{A}\mathbf{Q}_1\mathbf{Q}_2 \dots \mathbf{Q}_{n-1} = \mathbf{U}.$$

Если теперь воспользоваться формулой (1.18), то находим

$$(\mathbf{L}'_{n-1}\dots\mathbf{L}'_2\mathbf{L}'_1)(\mathbf{P}_{n-1}\dots\mathbf{P}_2\mathbf{P}_1)\mathbf{A}(\mathbf{Q}_1\mathbf{Q}_2\dots\mathbf{Q}_{n-1}) = \mathbf{U}.$$

Полагая $\mathbf{L} = (\mathbf{L}'_{n-1}\dots\mathbf{L}'_2\mathbf{L}'_1)^{-1}$, $\mathbf{P} = \mathbf{P}_{n-1}\dots\mathbf{P}_2\mathbf{P}_1$ и $\mathbf{Q} = \mathbf{Q}_1\mathbf{Q}_2\dots\mathbf{Q}_{n-1}$, приходим к равенству

$$\mathbf{PAQ} = \mathbf{LU}.$$

§ 5.7. Метод Холесского

В случае специальных матриц, к которым относятся симметрические, положительно определенные, разреженные, ленточные и некоторые другие виды матриц, метод исключения Гаусса может быть существенно оптимизирован как по числу выполняемых арифметических и логических операций, так и по используемой памяти. Весьма важным в приложениях является случай симметрических положительно определенных матриц, когда можно получить разложение матрицы на треугольные множители вдвое быстрее и с использованием вдвое меньшей памяти, чем в общем случае. Алгоритм такого разложения, известный как схема Холесского, является вариантом гауссова исключения, использующим и сохраняющим симметрию матрицы.

Напомним, что вещественная матрица $\mathbf{A}$ называется положительно определенной и обозначается символом $\mathbf{A} > 0$, если $\mathbf{x}^T\mathbf{Ax} > 0$ для всех векторов $\mathbf{x} \neq \mathbf{0}$.

Теорема 1.2. *Симметрическая положительно определенная матрица* $\mathbf{A}$ *обладает следующими свойствами:*

1. *если* $\mathbf{X}$ *– невырожденная матрица, то* $\mathbf{X}^T\mathbf{AX} > 0$;
2. *все главные миноры положительны*;
3. *все собственные значения положительны*;
4. *все диагональные элементы* $a_{ii} > 0$ *и* $\max_{ij}|a_{ij}| = \max_i a_{ii} > 0$.
5. *Если* $\mathbf{A} = \mathbf{LL}^T$, *где* $\mathbf{L}$ *– невырожденная матрица, то* $\mathbf{A} > 0$.

Доказательство 1. Так как матрица $\mathbf{X}$ невырождена, то $\mathbf{Xx} \neq 0$ для всех $\mathbf{x} \neq \mathbf{0}$. Поэтому $\mathbf{x}^T\mathbf{X}^T\mathbf{AXx} > 0$ при $\mathbf{x} \neq \mathbf{0}$. Таким образом, если $\mathbf{A} > 0$, то $\mathbf{X}^T\mathbf{AX} > 0$.

2. Пусть $\mathbf{A}_j = \mathbf{A}(1:j, 1:j)$. Тогда для всякого вектора $\mathbf{y}$ длины j вектор $\mathbf{x} = (\mathbf{y}^T, \mathbf{0})^T$ длины n удовлетворяет соотношению $\mathbf{y}^T\mathbf{A}_j\mathbf{y} = \mathbf{x}^T\mathbf{Ax}$. Поэтому, если $\mathbf{x}^T\mathbf{Ax} > 0$ для любого $\mathbf{x} \neq \mathbf{0}$, то $\mathbf{y}^T\mathbf{A}_j\mathbf{y} > 0$ для любого ненулевого вектора $\mathbf{y}$. Поэтому $\mathbf{A}_j > 0$.

3. Пусть $\mathbf{X}$ – вещественная ортогональная матрица, составленная из собственных векторов матрицы $\mathbf{A}$. Тогда $\mathbf{X}^T\mathbf{AX} = \mathbf{\Lambda}$ – диагональная матрица, имеющая на диагонали вещественные собственные числа λ_i. Поскольку $\mathbf{x}^T\mathbf{\Lambda x} = \sum_i \lambda_i x_i^2$, то $\Lambda > 0$, если и только если все $\lambda_i > 0$.

4. Пусть $\mathbf{e}_i$ – столбец с номером i единичной матрицы. Тогда $\mathbf{e}_i^T\mathbf{A}\mathbf{e}_i = a_{ii} > 0$ для всех i. Если $|a_{kl}| = \max_{i,j}|a_{ij}|$, но $k \neq l$, положим $\mathbf{x} = \mathbf{e}_k - \operatorname{sign}(a_{kl})\mathbf{e}_l$. Тогда $\mathbf{x}^T\mathbf{A}\mathbf{x} = a_{kk} + a_{ll} - 2|a_{kl}| \leq 0$, что противоречит условию $\mathbf{A} > 0$.

5. Пусть $\mathbf{A} = \mathbf{L}\mathbf{L}^T$, где $\mathbf{L}$ – невырожденная матрица. В этом случае получаем $\mathbf{x}^T\mathbf{A}\mathbf{x} = (\mathbf{x}^T\mathbf{L})(\mathbf{L}^T\mathbf{x}) = \|\mathbf{L}^T\mathbf{x}\|_2^2 > 0$ для всех $\mathbf{x} \neq \mathbf{0}$. Поэтому $\mathbf{A} > 0$. Теорема доказана.

Замечание 1.1. Вещественная несимметрическая матрица с положительными собственными значениями может не быть положительно определенной [6].

Покажем теперь, что если $\mathbf{A}$ – симметрическая положительно определенная матрица, то существует единственная невырожденная нижняя треугольная матрица $\mathbf{L}$ с положительными диагональными элементами такая, что

$$\mathbf{A} = \mathbf{L}\mathbf{L}^T. \tag{5.20}$$

Представление (1.20) называется *разложением Холесского* а матрица $\mathbf{L}$ – *множителем Холесского* матрицы $\mathbf{A}$.

Запишем разложение (1.20) в явном виде

$$\begin{pmatrix} l_{11} & 0 & \dots & 0 \\ l_{21} & l_{22} & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ l_{n1} & l_{n2} & \dots & l_{nn} \end{pmatrix} \begin{pmatrix} l_{11} & l_{12} & \dots & l_{1n} \\ 0 & l_{22} & \dots & l_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & l_{nn} \end{pmatrix} = \begin{pmatrix} a_{11} & a_{12} & \dots & a_{1n} \\ a_{21} & a_{22} & \dots & a_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ a_{n1} & a_{n2} & \dots & a_{nn} \end{pmatrix}.$$

Выполнив перемножение матриц, на основе поэлементного приравнивания левых и правых частей составим $n(n+1)/2$ уравнений относительно такого же числа неизвестных элементов матрицы $\mathbf{L}$:

$$\begin{array}{lllll} l_{11}^2 = a_{11}, & l_{11}l_{12} & = a_{12}, & \dots, & l_{11}l_{1n} & = a_{1n}, \\ & l_{12}^2 + l_{22}^2 = a_{22}, & \dots, & l_{12}l_{1n} + l_{22}l_{2n} & = a_{2n}, \\ & & \ddots & \vdots & \vdots & \\ & & & l_{1n}^2 + l_{2n}^2 + \dots + l_{nn}^2 & = a_{nn}. \end{array}$$

Из первой строки уравнений находим

$$l_{11} = \sqrt{a_{11}}, \quad l_{1j} = \frac{a_{1j}}{l_{11}}, \quad j = 2, 3, \dots, n.$$

Из второй строки получаем

$$l_{22} = \sqrt{a_{22} - l_{12}^2}, \quad l_{2j} = \frac{a_{2j} - l_{12}l_{1j}}{l_{22}}, \quad j = 3, 4, \dots, n$$

и т. д. Завершается процесс вычислением

$$l_{nn} = \sqrt{a_{nn} - \sum_{i=1}^{n-1} l_{in}^2}.$$

Таким образом, матрица $\mathbf{L}$ может быть вычислена по следующему *алгоритму Холесского*:

$$\begin{array}{l}
\texttt{for } j = 1 : n \\
\quad l_{jj} = (a_{jj} - \sum_{k=1}^{j-1} l_{jk}^2)^{1/2} \\
\quad\quad \texttt{for } i = j+1 : n \\
\quad\quad\quad l_{ij} = (a_{ij} - \sum_{k=1}^{j-1} l_{ik} l_{jk})/l_{jj} \\
\quad\quad \texttt{end} \\
\texttt{end}
\end{array} \tag{5.21}$$

Если $\mathbf{A}$ не является положительно определенной матрицей, то (в точной арифметике) алгоритм прекратит работу при попытке извлечь квадратный корень из отрицательного числа либо разделить на нуль. Это дает самый экономный способ проверки свойства положительной определенности симметрической матрицы.

Как и в случае гауссова исключения, $\mathbf{L}$ может быть записана на место элементов матрицы $\mathbf{A}$, расположенных на главной диагонали и ниже ее. В алгоритме происходит обращение только к этим элементам матрицы $\mathbf{A}$. Поэтому достаточно использовать $n(n+1)/2$ ячеек памяти вместо n^2. Число операций с плавающей запятой равно

$$\sum_{j=1}^{n} \Big(2j + \sum_{i=j+1}^{n} 2j\Big) = \frac{1}{3}n^3 + O(n^2),$$

т. е. примерно половина того, что выполняется при гауссовом исключении. Отметим также, что для численной устойчивости алгоритма Холесского выбор главных элементов не является необходимым [10].

При наличии $\mathbf{LL}^T$-разложения решение системы $\mathbf{Ax} = \mathbf{b}$ с симметрической положительно определенной матрицей сводится к последовательному решению двух треугольных систем

$$\mathbf{Ly} = \mathbf{b} \qquad \mathbf{L}^T\mathbf{x} = \mathbf{y}.$$

Остается воспользоваться формулами

$$\begin{aligned}
y_i &= \Big(b_i - \sum_{k=1}^{i-1} l_{ki} y_k\Big)/l_{ii}, \quad i = 1, 2, \ldots, n, \\
x_i &= \Big(y_i - \sum_{k=i+1}^{n} l_{ik} x_k\Big)/l_{ii}, \quad i = n, n-1, \ldots, 1.
\end{aligned} \tag{5.22}$$

Решение системы $\mathbf{Ax} = \mathbf{b}$ по формулам (1.21) и (1.22) называют *методом квадратного корня* или *методом Холесского*.

Сложнее обстоит дело с симметрическими, но незнакоопределенными матрицами. Можно показать [10], что для невырожденной матрицы $\mathbf{A}$ найдется матрица перестановок $\mathbf{P}$, нижняя треугольная матрица $\mathbf{L}$ и блочно-диагональная матрица

$\mathbf{D}$ с 1×1 и 2×2 диагональными блоками (комплексные числа представляются как пары вещественных чисел) такие, что $\mathbf{PAP}^T = \mathbf{LDL}^T$. Это разложение можно вычислить устойчиво, экономя при этом примерно половину операций и памяти по сравнению со стандартным гауссовым исключением.

В случае разреженных матриц весьма важно сохранить свойство разреженности за счет правильной организации вычислительного процесса.

Пример 1.6. Рассмотрим систему уравнений с симметрической положительно определенной матрицей [11]

$$\begin{pmatrix} 4 & 1 & 2 & \frac{1}{2} & 2 \\ 1 & \frac{1}{2} & 0 & 0 & 0 \\ 2 & 0 & 3 & 0 & 0 \\ \frac{1}{2} & 0 & 0 & \frac{5}{8} & 0 \\ 2 & 0 & 0 & 0 & 16 \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \\ x_5 \end{pmatrix} = \begin{pmatrix} 7 \\ 3 \\ 7 \\ -4 \\ -4 \end{pmatrix}.$$

Множитель Холесского для матрицы коэффициентов этой системы имеет вид

$$\mathbf{L} = \begin{pmatrix} 2 & 0 & 0 & 0 & 0 \\ 0,50 & 0,50 & 0 & 0 & 0 \\ 1 & -1 & 1 & 0 & 0 \\ 0,25 & -0,25 & -0,50 & 0,50 & 0 \\ 1 & -1 & -2 & -3 & 1 \end{pmatrix}.$$

Решая системы $\mathbf{Ly} = \mathbf{b}$ и $\mathbf{L}^T\mathbf{x} = \mathbf{y}$, получаем

$$\mathbf{y} = \begin{pmatrix} 3,5 \\ 2,5 \\ 6 \\ -2,5 \\ -0,50 \end{pmatrix}, \quad \mathbf{x} = \begin{pmatrix} 2 \\ 2 \\ 1 \\ -8 \\ -0,50 \end{pmatrix}.$$

Этот пример иллюстрирует важный факт, относящийся к применению метода Холесского для разреженной матрицы $\mathbf{A}$: матрица обычно претерпевает *заполнение*. Это означает, что $\mathbf{L}$ имеет ненулевые элементы в позициях, где в нижней треугольной части $\mathbf{A}$ стояли нули.

Перенумеруем теперь неизвестные по правилу $x_i \to \tilde{x}_{6-i}$, $i = 1, 2, \ldots, 5$ и переупорядочим уравнения по тому же закону. При этом последнее уравнение станет первым, предпоследнее вторым сверху и т. д. В результате получим следующую

эквивалентную систему уравнений

$$\begin{pmatrix} 16 & 0 & 0 & 0 & 2 \\ 0 & \frac{5}{8} & 0 & 0 & \frac{1}{2} \\ 0 & 0 & 3 & 0 & 2 \\ 0 & 0 & 0 & \frac{1}{2} & 1 \\ 2 & \frac{1}{2} & 2 & 1 & 4 \end{pmatrix} \begin{pmatrix} \tilde{x}_1 \\ \tilde{x}_2 \\ \tilde{x}_3 \\ \tilde{x}_4 \\ \tilde{x}_5 \end{pmatrix} = \begin{pmatrix} -4 \\ -4 \\ 7 \\ 3 \\ 7 \end{pmatrix}.$$

Ясно, что эта перенумерация переменных и переупорядочение уравнений равносильны симметрической перестановке строк и столбцов $\mathbf{A}$, причем та же перестановка применяется к $\mathbf{b}$. Эту новую систему запишем в виде $\tilde{\mathbf{A}}\tilde{\mathbf{x}} = \tilde{\mathbf{b}}$. Применяя к ней опять метод Холесского, разложим $\tilde{\mathbf{A}}$ в произведение $\tilde{\mathbf{L}}\tilde{\mathbf{L}}^T$, где (с точностью до трех значащих цифр)

$$\tilde{\mathbf{L}} = \begin{pmatrix} 4 & 0 & 0 & 0 & 0 \\ 0 & 0,791 & 0 & 0 & 0 \\ 0 & 0 & 1,73 & 0 & 0 \\ 0 & 0 & 0 & 0,707 & 0 \\ 0,500 & 0,632 & 1,15 & 1,41 & 0,129 \end{pmatrix}$$

Решая системы $\tilde{\mathbf{L}}\tilde{\mathbf{y}} = \tilde{\mathbf{b}}$ и $\tilde{\mathbf{L}}^T\tilde{\mathbf{x}} = \tilde{\mathbf{y}}$, получим вектор $\tilde{\mathbf{x}}$, который является всего лишь переупорядоченной формой $\mathbf{x}$. Таким образом, изменение порядка уравнений и неизвестных привело к треугольной матрице $\tilde{\mathbf{L}}$, которая разрежена в точности так же, как и нижняя треугольная часть $\mathbf{A}$.

На практике для большинства задач с разреженными матрицами разумное упорядочивание строк и столбцов матрицы коэффициентов дает огромное сокращение заполнения и, следовательно, экономию машинного времени и памяти. К таким задачам относятся, в частности, системы с ленточными матрицами, когда только диагональ, первые k поддиагоналей и l наддиагоналей содержат ненулевые элементы ($a_{ij} = 0$ для $i > j + k$ или $j > i + l$). В общем случае, применяя гауссово исключение без выбора ведущих элементов, получаем LU-разложение, где $\mathbf{L}$ и $\mathbf{U}$ – ленточные матрицы с ненулевыми диагональю, первыми k поддиагоналями и l наддиагоналями соответственно.

§ 5.8. Поведение числа обусловленности при матричных преобразованиях

При гауссовом исключении решение исходной системы сводится к решению системы с верхней треугольной матрицей $\mathbf{U} = \mathbf{L}^{-1}\mathbf{A}$. Появление перед $\mathbf{A}$ множителя $\mathbf{L}^{-1}$ может ухудшить обусловленность результирующей матрицы $\mathbf{U}$. Покажем, что если $\mathrm{cond}(\mathbf{L}^{-1})$ – большое число, то и $\mathrm{cond}(\mathbf{U})$ будет велико. Согласно

LU-разложению имеем $\mathbf{U} = \mathbf{L}^{-1}\mathbf{A}$ и $\mathbf{U}^{-1} = \mathbf{A}^{-1}\mathbf{L}$. Следовательно,

$$\|\mathbf{U}\| \le \|\mathbf{L}^{-1}\|\,\|\mathbf{A}\| \quad \text{и} \quad \|\mathbf{U}^{-1}\| \le \|\mathbf{A}^{-1}\|\,\|\mathbf{L}\|.$$

Перемножая эти неравенства, получаем оценку

$$\operatorname{cond}(\mathbf{U}) \le \operatorname{cond}(\mathbf{L}^{-1})\operatorname{cond}(\mathbf{A}).$$

Аналогично $\mathbf{L} = \mathbf{A}\mathbf{U}^{-1}$ и $\mathbf{L}^{-1} = \mathbf{U}\mathbf{A}^{-1}$ и справедливы неравенства

$$\|\mathbf{L}\| \le \|\mathbf{A}\|\,\|\mathbf{U}^{-1}\| \quad \text{и} \quad \|\mathbf{L}^{-1}\| \le \|\mathbf{U}\|\,\|\mathbf{A}^{-1}\|.$$

Поэтому $\operatorname{cond}(\mathbf{L}^{-1}) \le \operatorname{cond}(\mathbf{A})\operatorname{cond}(\mathbf{U})$ и в результате приходим к оценке

$$\frac{\operatorname{cond}(\mathbf{L}^{-1})}{\operatorname{cond}(\mathbf{A})} \le \operatorname{cond}(\mathbf{U}) \le \operatorname{cond}(\mathbf{A})\operatorname{cond}(\mathbf{L}^{-1}).$$

Число $\operatorname{cond}(\mathbf{A})$ задано и не зависит от метода решения исходной системы. Поэтому согласно полученной оценке при большом $\operatorname{cond}(\mathbf{L}^{-1})$ величина $\operatorname{cond}(\mathbf{U})$ также будет велика и обратный ход гауссова исключения окажется гораздо хуже обусловлен, чем исходная задача. На этом этапе может произойти большая потеря точности.

Пример 1.7. Рассмотрим матрицу $\mathbf{A}$, для которой LU-разложение имеет вид

$$\mathbf{A} = \begin{pmatrix} 1 & 0 & 0 & 1 \\ -1 & 1 & 0 & 1 \\ -1 & -1 & 1 & 1 \\ -1 & -1 & -1 & 1 \end{pmatrix} = \begin{pmatrix} 1 & 0 & 0 & 0 \\ -1 & 1 & 0 & 0 \\ -1 & -1 & 1 & 0 \\ -1 & -1 & -1 & 1 \end{pmatrix} \begin{pmatrix} 1 & 0 & 0 & 1 \\ 0 & 1 & 0 & 2 \\ 0 & 0 & 1 & 4 \\ 0 & 0 & 0 & 8 \end{pmatrix} = \mathbf{L}\mathbf{U}$$

или

$$\mathbf{U} = \begin{pmatrix} 1 & 0 & 0 & 1 \\ 0 & 1 & 0 & 2 \\ 0 & 0 & 1 & 4 \\ 0 & 0 & 0 & 8 \end{pmatrix} = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 \\ 2 & 1 & 1 & 0 \\ 4 & 2 & 1 & 1 \end{pmatrix} \begin{pmatrix} 1 & 0 & 0 & 1 \\ -1 & 1 & 0 & 1 \\ -1 & -1 & 1 & 1 \\ -1 & -1 & -1 & 1 \end{pmatrix} = \mathbf{L}^{-1}\mathbf{A}.$$

Аналогичная матрица $\mathbf{U}$ порядка $n \times n$ будет иметь в правом нижнем углу ведущий элемент $a_{nn}^{(n-1)} = 2^{n-1}$ а $\operatorname{cond}_1(\mathbf{L}^{-1}) = n2^{n-1}$, т. е. прямой ход гауссова исключения равносилен умножению на плохо обусловленную матрицу $\mathbf{L}^{-1}$. Для представления чисел с плавающей запятой типичный компьютер использует 64 двоичных разряда. Число решаемых уравнений может составлять сотни и тысячи. Поэтому потеря n разрядов представляется абсолютно неприемлемой. Таким образом, для некоторых видов матриц гауссово исключение с выбором ведущих элементов по столбцам является неустойчивым алгоритмом. Нужны алгоритмы получше!

Для устранения возникшего затруднения используют ортогональные матрицы. Напомним, что матрица $\mathbf{Q}$ называется ортогональной, если обратная матрица совпадает с транспонированной, т. е. $\mathbf{Q}\mathbf{Q}^T = \mathbf{Q}^T\mathbf{Q} = \mathbf{E}$. Примером таких матриц являются матрицы перестановок, использованные в гауссовом исключении с выбором ведущих элементов. Если получить разложение $\mathbf{A} = \mathbf{Q}\mathbf{R}$, где $\mathbf{Q}$ – ортогональная матрица а $\mathbf{R}$ – верхняя треугольная матрица, то решение исходной системы сведется к решению системы с верхней треугольной матрицей $\mathbf{R}\mathbf{x} = \mathbf{Q}^T\mathbf{b}$.

Покажем, что если $\mathbf{Q}$ – ортогональная матрица, то

$$\|\mathbf{Q}\mathbf{x}\|_2 = \|\mathbf{x}\|_2 \quad \text{и} \quad \|\mathbf{Q}\mathbf{A}\|_2 = \|\mathbf{A}\|_2.$$

Действительно,

$$\begin{aligned}
\|\mathbf{Q}\mathbf{x}\|_2^2 &= \mathbf{x}^T\mathbf{Q}^T\mathbf{Q}\mathbf{x} = \mathbf{x}^T\mathbf{x} = \|\mathbf{x}\|_2^2; \\
\|\mathbf{Q}\mathbf{A}\|_2^2 &= \max_{\|\mathbf{x}\|_2=1} \|\mathbf{Q}\mathbf{A}\mathbf{x}\|_2^2 = \max_{\|\mathbf{x}\|_2=1} \mathbf{x}^T\mathbf{A}^T\mathbf{Q}^T\mathbf{Q}\mathbf{A}\mathbf{x} = \\
&= \max_{\|\mathbf{x}\|_2=1} \|\mathbf{A}\mathbf{x}\|_2^2 = \|\mathbf{A}\|_2^2.
\end{aligned}$$

В качестве очевидного следствия отсюда получаем

$$\mathrm{cond}_2(\mathbf{Q}\mathbf{A}) = \|\mathbf{Q}\mathbf{A}\|_2 \, \|(\mathbf{Q}\mathbf{A})^{-1}\|_2 = \|\mathbf{A}\|_2 \|\mathbf{A}^{-1}\|_2 = \mathrm{cond}_2(\mathbf{A}),$$

т. е. при ортогональных преобразованиях обусловленность матрицы не меняется и $\mathrm{cond}_2(\mathbf{A}) = \mathrm{cond}_2(\mathbf{R})$. Поэтому представляется естественным использовать методы решения линейных систем, основанные на ортогональных преобразованиях. Существует целое семейство методов получения QR-разложения матрицы $\mathbf{A}$. Наиболее известными являются методы вращений, ортогонализации Грама-Шмидта и метод отражений.

§ 5.9. Метод вращений

Рассмотрим метод, в котором подобно гауссовому исключению последовательно зануляются поддиагональные элементы матрицы сначала в первом столбце, потом во втором и т. д. В результате матрица приводится к верхней треугольной форме. При этом, однако, обусловленность результирующей верхней треугольной матрицы не будет отличаться от обусловленности исходной матрицы. Таким образом, метод будет существенно более устойчив, чем гауссово исключение.

Пусть $a_{11} \neq 0$. На первом шаге исходная матрица $\mathbf{A}$ умножается на матрицу $\mathbf{Q}_{21}$, которая отличается от единичной матрицы только четырьмя элементами

в верхнем левом углу:

$$\mathbf{Q}_{21}\mathbf{A}=\begin{pmatrix} c & s & \dots & 0 \\ -s & c & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & 1 \end{pmatrix}\begin{pmatrix} a_{11} & a_{12} & \dots & a_{1n} \\ a_{21} & a_{22} & \dots & a_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ a_{n1} & a_{n2} & \dots & a_{nn} \end{pmatrix}=\begin{pmatrix} a_{11}^{(1)} & a_{12}^{(1)} & \dots & a_{1n}^{(1)} \\ a_{21}^{(1)} & a_{22}^{(1)} & \dots & a_{2n}^{(1)} \\ \vdots & \vdots & \ddots & \vdots \\ a_{n1} & a_{n2} & \dots & a_{nn} \end{pmatrix}$$

где

$$a_{1j}^{(1)} = ca_{1j} + sa_{2j}, \quad a_{2j}^{(1)} = -sa_{1j} + ca_{2j}, \quad j = 1, 2, \dots, n.$$

Если теперь положить

$$c = \frac{a_{11}}{\sqrt{a_{11}^2 + a_{21}^2}}, \quad s = \frac{a_{21}}{\sqrt{a_{11}^2 + a_{21}^2}},$$

то получаем $a_{21}^{(1)} = 0$. Так как $s^2 + c^2 = 1$, то величины s и c можно интерпретировать как $s = \sin\theta$ и $c = \cos\theta$ для некоторого угла поворота θ. При этом матрица $\mathbf{Q}_{21}$ оказывается ортогональной, и ее принято называть *матрицей вращения*.

На i-м шаге умножаем на матрицу $\mathbf{Q}_{i+1,1}$, которая отличается от единичной матрицы элементами: $\mathbf{Q}_{i+1,1}(1,1) = \mathbf{Q}_{i+1,1}(i,i) = c$, $\mathbf{Q}_{i+1,1}(1,i) = -\mathbf{Q}_{i+1,1}(i,1) = s$. Чтобы получить $a_{i1}^{(1)} = 0$, полагаем

$$c = \frac{a_{11}^{(i-1)}}{\sqrt{(a_{11}^{(i-1)})^2 + a_{i1}^2}}, \quad s = \frac{a_{i1}}{\sqrt{(a_{11}^{(i-1)})^2 + a_{i1}^2}}.$$

В результате выполнения $n-1$ шагов матрица $\mathbf{A}$ преобразуется к виду

$$\mathbf{Q}_{n,1}\mathbf{Q}_{n-1,1}\dots\mathbf{Q}_{2,1}\mathbf{A} = \begin{pmatrix} a_{11}^{(n-1)} & a_{12}^{(n-1)} & \dots & a_{1n}^{(n-1)} \\ 0 & a_{22}^{(1)} & \dots & a_{2n}^{(1)} \\ \vdots & \vdots & \ddots & \vdots \\ 0 & a_{n2}^{(1)} & \dots & a_{nn}^{(1)} \end{pmatrix}$$

Отметим, что первая строка была изменена $n-1$ раз.

Таким же образом исключаются поддиагональные элементы во втором и последующих столбцах. Всего нам нужно произвести умножение на $n-1+n-2+\ldots+1 = (n-1)n/2$ матриц вращения. Так как произведение ортогональных матриц дает опять ортогональную матрицу, то окончательно получаем

$$\mathbf{Q}_{n,n-1}\mathbf{Q}_{n,n-2}\dots\mathbf{Q}_{21}\mathbf{A} = \mathbf{Q}^T\mathbf{A} = \mathbf{R} \quad \text{или} \quad \mathbf{A} = \mathbf{QR},$$

где $\mathbf{R}$ – верхняя треугольная матрица. Это QR-разложение матрицы $\mathbf{A}$. Теперь решение системы $\mathbf{Ax} = \mathbf{b}$ сводится к решению системы с верхней треугольной матрицей $\mathbf{Rx} = \mathbf{Q}^T\mathbf{b}$. Последнее не отличается от обратного хода гауссова исключения.

Выше было показано, что при ортогональных преобразованиях обусловленность матрицы не меняется, т. е. $cond_2(\mathbf{A}) = cond_2(\mathbf{R})$. Поэтому метод вращений численно более устойчив, чем гауссово исключение. Этот метод требует, однако, проведения примерно $2n^3$ арифметических операций, что в три раза больше, чем при гауссовом исключении. Тем не менее среди методов QR-разложения, требующих для своей реализации $2n^3$ операций, *метод вращений* рассматривается как наиболее устойчивый к вычислительной погрешности.

Пример 1.8. Рассмотрим решение методом вращений системы

$$\begin{pmatrix} 1 & 2 \\ 1 & 3 \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \end{pmatrix} = \begin{pmatrix} 1 \\ 2 \end{pmatrix}. \tag{5.23}$$

Умножение на матрицу вращений зануляет элемент в позиции (2,1):

$$\mathbf{Q} \begin{pmatrix} 1 \\ 1 \end{pmatrix} = \begin{pmatrix} c & s \\ -s & c \end{pmatrix} \begin{pmatrix} 1 \\ 1 \end{pmatrix} = \begin{pmatrix} * \\ 0 \end{pmatrix}.$$

Отсюда получаем $-s + c = 0$ или $s = c = 1/\sqrt{2}$. Поэтому

$$\begin{aligned} \mathbf{QA} &= \frac{1}{\sqrt{2}} \begin{pmatrix} 1 & 1 \\ -1 & 1 \end{pmatrix} \begin{pmatrix} 1 & 2 \\ 1 & 3 \end{pmatrix} = \frac{1}{\sqrt{2}} \begin{pmatrix} 2 & 5 \\ 0 & 1 \end{pmatrix} = \mathbf{R}, \\ \mathbf{Qb} &= \frac{1}{\sqrt{2}} \begin{pmatrix} 1 & 1 \\ -1 & 1 \end{pmatrix} \begin{pmatrix} 1 \\ 2 \end{pmatrix} = \frac{1}{\sqrt{2}} \begin{pmatrix} 3 \\ 1 \end{pmatrix}. \end{aligned}$$

В результате приходим к системе с верхней треугольной матрицей

$$\begin{pmatrix} 2 & 5 \\ 0 & 1 \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \end{pmatrix} = \begin{pmatrix} 3 \\ 1 \end{pmatrix},$$

откуда обратной подстановкой получаем решение $x_2 = 1$, $x_1 = -1$.

§ 5.10. Метод ортогонализации Грама-Шмидта

Этот метод основан на разложении матрицы коэффициентов системы $\mathbf{Ax} = \mathbf{b}$ в произведение ортогональной и верхней треугольной матриц. Будем считать, что матрица $\mathbf{A}$ невырождена и, следовательно, ее столбцы $\mathbf{a}_i$, $i = 1, 2, \ldots, n$ линейно независимы. Идея QR-разложения матрицы $\mathbf{A}$ в произведение ортогональной и верхней треугольной матриц состоит в том, что столбцы матрицы $\mathbf{Q}$, т. е. ортонормированные вектора $\mathbf{q}_i$, $i = 1, 2, \ldots, n$, должны образовывать то же пространство, что и столбцы матрицы $\mathbf{A}$.

Равенство $\mathbf{A} = \mathbf{QR}$ запишем в виде

$$\left(\begin{array}{c|c|c|c} & & & \\ \mathbf{a}_1 & \mathbf{a}_2 & \dots & \mathbf{a}_n \\ & & & \end{array}\right) = \left(\begin{array}{c|c|c|c} & & & \\ \mathbf{q}_1 & \mathbf{q}_2 & \dots & \mathbf{q}_n \\ & & & \end{array}\right) \begin{pmatrix} r_{11} & r_{12} & \dots & r_{1n} \\ & r_{22} & \dots & r_{2n} \\ & & \ddots & \vdots \\ & & & r_{nn} \end{pmatrix}, \tag{5.24}$$

где диагональные элементы r_{ii} отличны от нуля. Отсюда следует, что векторы $\mathbf{a}_1, \dots, \mathbf{a}_i$ могут быть выражены как линейные комбинации $\mathbf{q}_1, \dots, \mathbf{q}_i$ и обратно при условии обратимости верхних $i \times i$ блоков матрицы $\mathbf{R}$.

Равенство (1.24) дает уравнения

$$\begin{aligned} \mathbf{a}_1 &= r_{11}\mathbf{q}_1, \\ \mathbf{a}_2 &= r_{12}\mathbf{q}_1 + r_{22}\mathbf{q}_2, \\ &\dots \\ \mathbf{a}_n &= r_{1n}\mathbf{q}_1 + r_{2n}\mathbf{q}_2 + \dots + r_{nn}\mathbf{q}_n, \end{aligned}$$

которые можно переписать в виде

$$\begin{aligned} \mathbf{q}_1 &= \frac{\mathbf{a}_1}{r_{11}}, \\ \mathbf{q}_2 &= \frac{\mathbf{a}_2 - r_{12}\mathbf{q}_1}{r_{22}}, \\ &\dots \\ \mathbf{q}_n &= \frac{\mathbf{a}_n - \sum_{i=1}^{n-1} r_{in}\mathbf{q}_i}{r_{nn}}. \end{aligned} \tag{5.25}$$

Остается определить коэффициенты r_{ij}. Алгоритм построения по заданному набору линейно независимых векторов $\mathbf{a}_1, \mathbf{a}_2, \dots, \mathbf{a}_n$ системы ортонормированных векторов $\mathbf{q}_1, \mathbf{q}_2, \dots, \mathbf{q}_n$ известен как *классический процесс ортогонализации Грама-Шмидта*. Согласно этому алгоритму, исходя из требования

$$\mathbf{q}_i^T \mathbf{q}_j = 0, \quad i \neq j, \quad \|\mathbf{q}_j\|_2 = 1,$$

полагаем

$$r_{ij} = \mathbf{q}_i^T \mathbf{a}_j, \quad i \neq j, \quad r_{jj} = \|\mathbf{a}_j - \sum_{i=1}^{j-1} r_{ij}\mathbf{q}_i\|_2. \tag{5.26}$$

Таким образом, матрицы $\mathbf{Q}$ и $\mathbf{R}$ могут быть вычислены по следующему алгоритму, где на j-м шаге формируются j-е столбцы этих матриц:

```
for j = 1 : n
    v_j = a_j
        for i = 1 : j − 1
            r_ij = q_i^T a_j
            v_j = v_j − r_ij q_i
        end
    r_jj = ‖v_j‖_2
    q_j = v_j / r_jj
end
```

К сожалению, в условиях наличия ошибок округления этот алгоритм является численно неустойчивым. При проведении вычислений по формулам (1.25), (1.26) векторы $\mathbf{q}_i$ могут оказаться далеко не ортогональными, причем искажающее влияние ошибок округления будет тем больше, чем ближе столбцы $\mathbf{a}_i$ к линейно зависимым. Изменение порядка вычислений, известное как *модифицированный метод ортогонализации Грама-Шмидта*, позволяет избежать этого затруднения. Отличие состоит в том, что на j-м шаге модифицированного алгоритма формируется j-й столбец матрицы $\mathbf{Q}$ и j-я строка матрицы $\mathbf{R}$:

```
for i = 1 : n   v_i = a_i  end
for i = 1 : n
    r_ii = ‖v_i‖_2
    q_i = v_i / r_ii
        for j = i + 1 : n
            r_ij = q_i^T v_j
            v_j = v_j − r_ij q_i
        end
end
```

Внутренний цикл требует примерно $4n$ арифметических операций. Общее число арифметических операций (здесь и в классическом процессе Грама-Шмидта) асимптотически равно

$$\sum_{i=1}^{n}\sum_{j=i+1}^{n} 4n = \sum_{i=1}^{n}(i-1)4n \approx 2n^3.$$

Каждый внешний шаг модифицированного алгоритма Грама-Шмидта может быть интерпретирован как умножение матрицы $\mathbf{A}$ справа на верхнюю треугольную матрицу, отличающуюся от единичной только одной строкой. На первом шаге столбец $\mathbf{v}_1 = \mathbf{a}_1$ умножается на $1/r_{11}$, и полученный результат $\mathbf{q}_1$ вычитается из каждого последующего столбца $\mathbf{v}_j = \mathbf{a}_j$ с множителем r_{1j}. Это эквивалентно

умножению справа на матрицу $\mathbf{R}_1$:

$$\left(\begin{array}{c|c|c|c} & & & \\ \mathbf{v}_1 & \mathbf{v}_2 & \dots & \mathbf{v}_n \\ & & & \end{array}\right)\begin{pmatrix} \frac{1}{r_{11}} & \frac{-r_{12}}{r_{11}} & \dots & \frac{-r_{1n}}{r_{11}} \\ & 1 & \dots & 0 \\ & & \ddots & \vdots \\ & & & 1 \end{pmatrix} = \left(\begin{array}{c|c|c|c} & & & \\ \mathbf{q}_1 & \mathbf{v}_2^{(1)} & \dots & \mathbf{v}_n^{(1)} \\ & & & \end{array}\right).$$

На i-м шаге алгоритма происходит умножение справа на матрицу

$$\mathbf{R}_i = \begin{pmatrix} 1 & & & & & \\ & \ddots & & & & \\ & & \frac{1}{r_{ii}} & \frac{-r_{ii+1}}{r_{ii}} & \dots & \frac{-r_{in}}{r_{ii}} \\ & & & 1 & \dots & 0 \\ & & & & \ddots & \vdots \\ & & & & & 1 \end{pmatrix}.$$

Нетрудно заметить сходство матриц $\mathbf{R}_i$ с матрицами $\mathbf{L}_j$ в гауссовом исключении. Матрицы $\mathbf{R}_i$ обладают следующими свойствами:

1) обратная матрица $\mathbf{R}_i^{-1}$ получается из $\mathbf{R}_i$ заменой i-й строки на строку $(0; \dots; 0; r_{ii}; r_{ii+1}; \dots; r_{in})$;

2) произведение матриц $\mathbf{R}_{i+1}^{-1}\mathbf{R}_i^{-1}$ дает верхнюю треугольную матрицу, отличающуюся от единичной матрицы наличием строк i и и $i+1$ матриц $\mathbf{R}_i^{-1}$ и $\mathbf{R}_{i+1}^{-1}$ на их обычных местах.

Окончательно получаем равенство

$$\mathbf{A}\mathbf{R}_1\mathbf{R}_2 \dots \mathbf{R}_n = \mathbf{Q}$$

или

$$\mathbf{A} = \mathbf{Q}\mathbf{R}_n^{-1}\mathbf{R}_{n-1}^{-1} \dots \mathbf{R}_1^{-1} = \mathbf{Q}\mathbf{R}.$$

Теперь решение линейной системы $\mathbf{A}\mathbf{x} = \mathbf{b}$ сводится к решению системы с верхней треугольной матрицей

$$\mathbf{R}\mathbf{x} = \mathbf{Q}^T\mathbf{b}.$$

В целом метод требует примерно $2n^3$ арифметических операций, что сравнимо с методом вращений, но больше, чем при гауссовом исключении, где используется примерно $2n^3/3$ операций. Модифицированный метод Грама-Шмидта существенно более устойчив. Однако, если $\mathbf{A}$ плохо обусловлена, то матрица $\mathbf{Q}$ может сильно отличаться от ортогональной, т. е. величина $||\mathbf{Q}^T\mathbf{Q} - \mathbf{E}||_2$ не будет достаточно мала.

Пример 1.9. Рассмотрим решение системы (1.23) модифицированным методом ортогонализации Грама-Шмидта. Согласно этому алгоритму здесь полагаем $\mathbf{v}_1 = \mathbf{a}_1 = (1;1)^T$, $\mathbf{v}_2 = \mathbf{a}_2 = (2;3)^T$. При $i = 1$ получаем

$$\begin{aligned} r_{11} &= \|\mathbf{v}_1\|_2 = \sqrt{2}, \quad \mathbf{q}_1 = \mathbf{v}_1/r_{11} = (1/\sqrt{2}; 1/\sqrt{2})^T, \\ r_{12} &= \mathbf{q}_1^T\mathbf{v}_2 = 5/\sqrt{2}, \quad \mathbf{v}_2 = \mathbf{v}_2 - r_{12}\mathbf{q}_1 = (-1/2; 1/2)^T. \end{aligned}$$

При $i = 2$ дополнительно находим

$$r_{22} = \|\mathbf{v}_2\|_2 = 1/\sqrt{2}, \quad \mathbf{q}_2 = \mathbf{v}_2/r_{22} = (-1/\sqrt{2}; 1/\sqrt{2})^T.$$

Таким образом, искомое QR-разложение матрицы $\mathbf{A}$ имеет вид

$$\mathbf{A} = \begin{pmatrix} 1 & 2 \\ 1 & 3 \end{pmatrix} = \frac{1}{\sqrt{2}} \begin{pmatrix} 1 & -1 \\ 1 & 1 \end{pmatrix} \frac{1}{\sqrt{2}} \begin{pmatrix} 2 & 5 \\ 0 & 1 \end{pmatrix} = \mathbf{QR}.$$

Умножая исходную систему (1.23) слева на матрицу $\mathbf{Q}^T$, получаем

$$\mathbf{Rx} = \mathbf{Q}^T\mathbf{b} \quad \text{или} \quad \begin{pmatrix} 2 & 5 \\ 0 & 1 \end{pmatrix} \begin{pmatrix} x_1 \\ rx_2 \end{pmatrix} = \begin{pmatrix} 3 \\ 1 \end{pmatrix}.$$

Отсюда $x_2 = 1$ и $x_1 = -1$.

§ 5.11. Метод отражений

Этот метод основан на получении QR-разложения матрицы $\mathbf{A}$ путем последовательного применения к ней ортогональных преобразований

$$\mathbf{Q}_{n-1}\mathbf{Q}_{n-2}\ldots\mathbf{Q}_1\mathbf{A} = \mathbf{R}.$$

Произведение ортогональных матриц $\mathbf{Q} = \mathbf{Q}_1^T\mathbf{Q}_2^T\ldots\mathbf{Q}_{n-1}^T$ также является ортогональной матрицей, что позволяет получить разложение $\mathbf{A} = \mathbf{QR}$. На j-м шаге алгоритма зануляются поддиагональные элементы j-го столбца с сохранением ранее полученных нулей в предшествующих столбцах:

$$\mathbf{A} = \begin{pmatrix} * & * & * & * \\ * & * & * & * \\ * & * & * & * \\ * & * & * & * \end{pmatrix} \overset{\mathbf{Q}_1}{\to} \begin{pmatrix} * & * & * & * \\ 0 & * & * & * \\ 0 & * & * & * \\ 0 & * & * & * \end{pmatrix} \overset{\mathbf{Q}_2}{\to} \begin{pmatrix} * & * & * & * \\ 0 & * & * & * \\ 0 & 0 & * & * \\ 0 & 0 & * & * \end{pmatrix} \overset{\mathbf{Q}_3}{\to} \begin{pmatrix} * & * & * & * \\ 0 & * & * & * \\ 0 & 0 & * & * \\ 0 & 0 & 0 & * \end{pmatrix}.$$

Каждая из матриц $\mathbf{Q}_j$ имеет вид $\mathbf{Q}_j = \begin{pmatrix} \mathbf{E} & \mathbf{0} \\ \mathbf{0} & \mathbf{H} \end{pmatrix}$, где $\mathbf{E}$ – единичная матрица размера $(j-1)\times(j-1)$ а $\mathbf{H}$ – ортогональная матрица размера $(n-j+1)\times(n-j+1)$. Очевидно, что $\mathbf{Q}$ – ортогональная матрица. Умножение на нее сохраняет первые

$j-1$ строк и столбцов и должно дать нули в позициях $j+1,\ldots,n$ в j-ом столбце преобразуемой матрицы.

В качестве нужной нам матрицы $\mathbf{H}$ используем матрицу

$$\mathbf{H} = \mathbf{E} - 2\mathbf{w}\mathbf{w}^T,$$

где вектор-столбец $\mathbf{w}$ имеет единичную длину, т. е. $\|\mathbf{w}\|_2^2 = \mathbf{w}^T\mathbf{w} = 1$. Матрица $\mathbf{H}$ является симметрической и ортогональной. Действительно,

$$\begin{aligned}\mathbf{H}^T &= (\mathbf{E} - 2\mathbf{w}\mathbf{w}^T)^T = \mathbf{E}^T - 2(\mathbf{w}\mathbf{w}^T)^T = \mathbf{E} - 2\mathbf{w}\mathbf{w}^T,\\ \mathbf{H}^T\mathbf{H} &= (\mathbf{E} - 2\mathbf{w}\mathbf{w}^T)(\mathbf{E} - 2\mathbf{w}\mathbf{w}^T) = \mathbf{E} - 4\mathbf{w}\mathbf{w}^T + 4\mathbf{w}\mathbf{w}^T\mathbf{w}\mathbf{w}^T = \mathbf{E}.\end{aligned}$$

Вектор $\mathbf{w}$ является собственным вектором матрицы $\mathbf{H}$ с собственным значением -1:

$$\mathbf{H}\mathbf{w} = (\mathbf{E} - 2\mathbf{w}\mathbf{w}^T)\mathbf{w} = -\mathbf{w}.$$

Всякий вектор $\mathbf{v}$, ортогональный вектору $\mathbf{w}$, также является собственным вектором для $\mathbf{H}$ с собственным числом $+1$:

$$\mathbf{H}\mathbf{v} = (\mathbf{E} - 2\mathbf{w}\mathbf{w}^T)\mathbf{v} = \mathbf{v} - 2\mathbf{w}\mathbf{w}^T\mathbf{v} = \mathbf{v}.$$

Множество векторов $\mathbf{v}$, ортогональных вектору $\mathbf{w}$, обозначим через G. Оно является гиперплоскостью, т. е. подпространством, размерность которого на единицу меньше размерности основного пространства.

Представим вектор $\mathbf{x}$ в виде суммы $\mathbf{x} = \mathbf{y} + \mathbf{v}$, где $\mathbf{y} = \alpha\mathbf{w}$ и $\mathbf{v}^T\mathbf{w} = 0$. Для этого следует взять в качестве $\mathbf{y}$ проекцию вектора $\mathbf{x}$ на вектор $\mathbf{w}$, т. е. $\mathbf{y} = (\mathbf{w}^T\mathbf{x})\mathbf{w}$ и $\mathbf{v} = \mathbf{x} - (\mathbf{w}^T\mathbf{x})\mathbf{w}$. Так как $\mathbf{H}\mathbf{x} = -\mathbf{y} + \mathbf{v}$, то $\mathbf{H}\mathbf{x}$ есть зеркальное отражение вектора $\mathbf{x}$ относительно гиперплоскости G, ортогональной вектору $\mathbf{w}$. По этой причине матрицу $\mathbf{H}$ называют *матрицей отражения.*

Предположим, что в начале j-го шага преобразуемая часть j-го столбца есть вектор $\mathbf{x} \in \mathbb{R}^{n-j+1}$. Так как при ортогональных преобразованиях длины векторов сохраняются, то

$$\mathbf{H}\mathbf{x} = (\mathbf{E} - 2\mathbf{w}\mathbf{w}^T)\mathbf{x} = \mathbf{x} - 2\mathbf{w}\mathbf{w}^T\mathbf{x} = \sigma\,\mathbf{e}_1, \quad \sigma = \pm\|\mathbf{x}\|_2,$$

где $\mathbf{e}_1 = (1, 0, \ldots, 0)^T \in \mathbb{R}^{n-j+1}$. Таким образом, получаем

$$2(\mathbf{w}^T\mathbf{x})\mathbf{w} = \mathbf{x} - \sigma\,\mathbf{e}_1.$$

Отсюда следует, что вектор $\mathbf{w}$ отличается от вектора $\mathbf{x} - \sigma\,\mathbf{e}_1$ только множителем, и так как вектор $\mathbf{w}$ имеет единичную длину, то

$$\mathbf{w} = \frac{\mathbf{x} - \sigma\,\mathbf{e}_1}{\|\mathbf{x} - \sigma\,\mathbf{e}_1\|_2}.$$

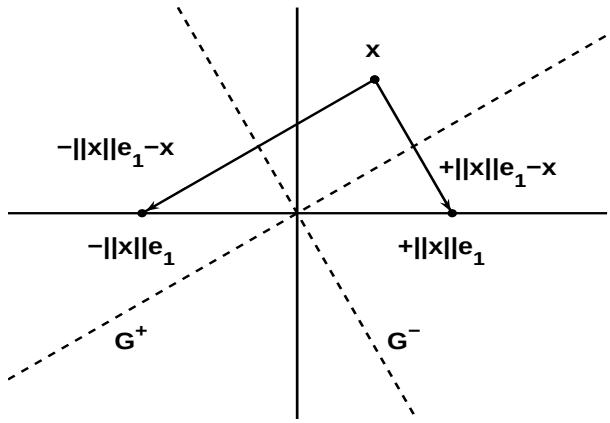


Рис. 5.1. Два возможных отражения. Для численной устойчивости важно выбрать то из них, которой сдвигает $\mathbf{x}$ на большее расстояние

Знак коэффициента σ выбирается из геометрических соображений таким образом, чтобы не допустить малость длины вектора $\mathbf{w}$ и, как следствие, обеспечить бóльшую устойчивость метода по отношению к ошибкам округления. Для этого достаточно положить $\sigma = -\mathrm{sign}(x_1)\|\mathbf{x}\|_2$, где x_1 – первая компонента вектора $\mathbf{x}$. Тогда

$$\mathbf{w} = \frac{\mathbf{z}}{\|\mathbf{z}\|_2}, \quad \mathbf{z} = \mathbf{x} + \mathrm{sign}(x_1)\|\mathbf{x}\|_2\,\mathbf{e}_1.$$

Для корректности определения положим $\mathrm{sign}(x_1) = 1$ при $x_1 = 0$.

Идея такого выбора иллюстрируется на рис. 1.1. Пусть G^+ и G^- – гиперплоскости (видимые «с грани», относительно которых проводится отражение. Предположим, что угол между G^+ и осью $\mathbf{e}_1$ очень мал. Тогда вектор $\mathbf{z}$ будет много меньше, чем $\mathbf{x}$ или $\|\mathbf{x}\|_2\,\mathbf{e}_1$. Таким образом, при вычислении $\mathbf{z}$ будут вычитаться близкие величины и результат будет подвержен ошибкам округления. Выбирая знак так, как это сделано выше, мы устраняем этот эффект, поскольку в этом случае $\|\mathbf{z}\|_2$ никогда не будет меньше, чем $\|\mathbf{x}\|_2$. Если векторы $\mathbf{x}$ и $\mathbf{e}_1$ коллинеарны (в частности, если все компоненты вектора $\mathbf{x}$ равны нулю), то отражение проводить не надо и мы сразу переходим к следующему шагу. Поэтому данный алгоритм всегда реализуем.

Для системы $\mathbf{Ax} = \mathbf{b}$ обозначим через $A[k:n, j]$ и $b[k:n]$ элементы j-го столбца матрицы $\mathbf{A}$ и вектора правой части $\mathbf{b}$ с номерами от k до n. Приводимый ниже алгоритм позволяет вычислить матрицу и правую часть линейной системы $\mathbf{Rx} = \mathbf{Q}^T\mathbf{b}$. При этом матрица $\mathbf{R}$ помещается на место матрицы $\mathbf{A}$.

```
for k = 1 : n − 1
    x = A[k : n, k]
    v_k = sign(x_1)‖x‖_2 e_1 + x
    v_k = v_k/‖v_k‖_2
    b[k : n] = b[k : n] − 2v_k(v_k^T b[k : n])
        for j = k : n
            A[k : n, j] = A[k : n, j] − 2v_k(v_k^T A[k : n, j])
        end
end
```

Данный алгоритм требует для своей реализации примерно $4n^3/3$ арифметических операций, что меньше, чем в методах вращений и ортогонализации Грама-Шмидта. Здесь также $\mathrm{cond}(\mathbf{R}) = \mathrm{cond}(\mathbf{A})$, что не ухудшает численную устойчивость метода. Более того, по объему используемой памяти, числу выполняемых операций и устойчивости метод отражений является одним из лучших алгоритмов для решения систем линейных алгебраических уравнений.

Пример 1.10. Рассмотрим решение методом отражений системы (1.23). В нашем случае

$$\mathbf{z} = \mathbf{x} + \mathrm{sign}(x_1)\|\mathbf{x}\|_2\,\mathbf{e}_1 = \begin{pmatrix}1\\1\end{pmatrix} + \sqrt{2}\begin{pmatrix}1\\0\end{pmatrix} = \begin{pmatrix}1+\sqrt{2}\\1\end{pmatrix},\ \|\mathbf{z}\|_2^2 = 4+2\sqrt{2},$$

$$\mathbf{w} = \frac{\mathbf{z}}{\|\mathbf{z}\|_2} = \frac{1}{\sqrt{4+2\sqrt{2}}}\begin{pmatrix}1+\sqrt{2},\\1\end{pmatrix},\quad \mathbf{H} = \mathbf{E} - 2\mathbf{w}\mathbf{w}^T = -\frac{1}{\sqrt{2}}\begin{pmatrix}1 & 1\\1 & -1\end{pmatrix}.$$

Поскольку умножение на матрицу отражений $\mathbf{H}$ дает

$$\mathbf{HA} = -\frac{1}{\sqrt{2}}\begin{pmatrix}1 & 1\\1 & -1\end{pmatrix}\begin{pmatrix}1 & 1\\2 & 3\end{pmatrix} = -\frac{1}{\sqrt{2}}\begin{pmatrix}2 & 5\\0 & -1\end{pmatrix},\quad \mathbf{Hb} = -\frac{1}{\sqrt{2}}\begin{pmatrix}3\\-1\end{pmatrix},$$

то приходим к системе

$$\begin{pmatrix}2 & 5\\0 & -1\end{pmatrix}\begin{pmatrix}x_1\\x_2\end{pmatrix} = \begin{pmatrix}3\\-1\end{pmatrix}.$$

Отсюда $x_2 = 1$ и $x_1 = -1$.

Отметим, что матрицу отражений $\mathbf{H}$ можно записать в виде

$$\mathbf{H} = \mathbf{E} - 2\mathbf{w}\mathbf{w}^T = \begin{pmatrix}-c & s\\s & c\end{pmatrix},$$

где $c = \cos(-\pi/4)$, $s = \sin(-\pi/4)$.

§ 5.12. Метод наименьших квадратов

Рассмотрим использование ортогональных преобразований при решении задач метода наименьших квадратов. Пусть имеются данные (x_i, y_i), $i = 0, 1, \ldots, N$ и требуется найти многочлен $S(x) = \sum_{j=1}^{M} c_j x^{j-1}$ степени $M-1$ такой, что минимизируется величина

$$E_M(c_1, c_2, \ldots, c_M) = \sum_{i=0}^{N} \Big[y_i - \sum_{j=1}^{M} c_j x^{j-1} \Big]^2.$$

Использование стандартного метода наименьших квадратов дает систему нормальных уравнений, решение которой доставляет коэффициенты искомого многочлена. Однако матрица системы нормальных уравнений является плохообусловленной и малые изменения входных данных приводят к сильному изменению решения. Поэтому такой путь получения решения нежелателен. Рассмотрим применение QR-факторизации для решения линейных систем метода наименьших квадратов.

Пусть требуется решить задачу поиска минимума

$$\min \|\mathbf{b} - \mathbf{A}\mathbf{c}\|_2^2,$$

где

$$\mathbf{A} = \begin{pmatrix} 2 & 3 & 1 & 2 \\ 0 & 2 & 4 & 1 \\ 0 & 0 & 1 & 2 \\ 0 & 0 & 0 & 5 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{pmatrix}, \quad \mathbf{b} = \begin{pmatrix} 1 \\ 2 \\ 3 \\ 10 \\ 5 \\ 2 \\ 3 \end{pmatrix}.$$

Так как три последние строки матрицы $\mathbf{A}$ состоят из нулей, то задача сводится к следующей

$$\min \|\mathbf{b}_1 - \mathbf{R}\mathbf{c}\|_2^2 + \|\mathbf{b}_2\|_2^2,$$

где

$$\mathbf{R} = \begin{pmatrix} 2 & 3 & 1 & 2 \\ 0 & 2 & 4 & 1 \\ 0 & 0 & 1 & 2 \\ 0 & 0 & 0 & 5 \end{pmatrix}, \quad \mathbf{b}_1 = \begin{pmatrix} 1 \\ 2 \\ 3 \\ 10 \end{pmatrix}, \quad \mathbf{b}_2 = \begin{pmatrix} 5 \\ 2 \\ 3 \end{pmatrix}.$$

Отметим, что вектор $\mathbf{b}_2$ не влияет на $\mathbf{c}$ и не участвует в поиске минимума. Слагаемое $\|\mathbf{b}_1 - \mathbf{R}\mathbf{c}\|_2^2$ можно занулить, решая систему линейных уравнений с треугольной матрицей $\mathbf{R}\mathbf{c} = \mathbf{b}_1$, что дает $\mathbf{c}^T = (-4; 2; -1; 2)$.

Таким образом, если в задаче МНК матрицу $\mathbf{A}$ порядка $N \times M$ системы линейных уравнений можно привести к верхней треугольной форме, то далее эту систему легко решить, используя обратную подстановку. Покажем, что для этого нельзя воспользоваться методом исключения Гаусса, так как элементарные операции над строками матрицы не сохраняют ее евклидову норму.

Пример 1.11. Пусть имеются точки $(1;1)$, $(2;1,2)$, $(3;1,5)$, $(4;1,8)$ и требуется найти прямую линию, обеспечивающую наилучшее приближение этих данных в смысле метода наименьших квадратов. Полагая

$$\mathbf{A} = \begin{pmatrix} 1 & 1 \\ 1 & 2 \\ 1 & 3 \\ 1 & 4 \end{pmatrix}, \quad \mathbf{b} = \begin{pmatrix} 1,0 \\ 1,2 \\ 1,5 \\ 1,8 \end{pmatrix},$$

и решая систему нормальных уравнений, находим $y = 0,70 + 0,27x$, причем $\|\mathbf{b} - \mathbf{Ac}\|_2^2 = 0,003$. Применим теперь метод гауссова исключения к расширенной матрице, которая преобразуется следующим образом

$$\left(\begin{array}{cc|c} 1 & 1 & 1,0 \\ 1 & 2 & 1,2 \\ 1 & 3 & 1,5 \\ 1 & 4 & 1,8 \end{array}\right) \to \left(\begin{array}{cc|c} 1 & 1 & 1,0 \\ 0 & 1 & 0,2 \\ 1 & 2 & 0,5 \\ 1 & 3 & 0,8 \end{array}\right) \to \left(\begin{array}{cc|c} 1 & 1 & 1,0 \\ 0 & 1 & 0,2 \\ 0 & 0 & 0,1 \\ 0 & 0 & 0,2 \end{array}\right).$$

Это дает прямую $y = 0,8 + 0,2x$, которая отличается от полученной по методу наименьших квадратов в худшую сторону. Теперь $\|\tilde{\mathbf{b}} - \mathbf{A}\tilde{\mathbf{c}}\|_2^2 = 0,05$, где $\tilde{\mathbf{b}}$ – новая правая часть, $\tilde{\mathbf{A}}$ – новая матрица коэффициентов и $\tilde{\mathbf{c}}$ – новый вектор параметров.

Обратимся теперь к QR-разложению прямоугольной матрицы $\mathbf{A}$. Воспользуемся тем фактом, что для всякой ортогональной матрицы $\mathbf{Q}$ имеет место равенство

$$\|\mathbf{b} - \mathbf{Ac}\|_2^2 = \|\mathbf{Q}(\mathbf{b} - \mathbf{Ac})\|_2^2 = \|\mathbf{Qb} - (\mathbf{QA})\mathbf{c})\|_2^2.$$

Поэтому минимизация $\|\mathbf{b} - \mathbf{Ac}\|_2^2$ может быть выполнена путем минимизации $\|\mathbf{Qb} - (\mathbf{QA})\mathbf{c}\|_2^2$. Более того, если равенство $\mathbf{QA} = \mathbf{R}$ дает верхнюю треугольную матрицу, то задача МНК может быть сведена к обратной подстановке гауссова исключения. Найденное решение будет также решением исходной задачи. Для получения нужной ортогональной матрицы $\mathbf{Q}$ используем матрицы отражения.

Пример 1.12. Рассмотрим опять задачу построения наилучшей в смысле МНК прямой, приближающей данные $(1;1)$, $(2;1,2)$, $(3;1,5)$, $(4;1,8)$. Требуется

построить ортогональную матрицу $\mathbf{Q}$, которая преобразует матрицу

$$\mathbf{A} = \begin{pmatrix} 1 & 1 \\ 1 & 2 \\ 1 & 3 \\ 1 & 4 \end{pmatrix}$$

в верхнюю треугольную матрицу. Сделаем это за два шага. На первом шаге получим матрицу отражения $\mathbf{Q}_1$, которая зануляет элементы ниже главной диагонали в первом столбце матрицы $\mathbf{A}$. Имеем

$$\mathbf{Q}_1 = \frac{1}{6}\begin{pmatrix} -3 & -3 & -3 & -3 \\ -3 & 5 & -1 & -1 \\ -3 & -1 & 5 & -1 \\ -3 & -1 & -1 & 5 \end{pmatrix}, \quad \mathbf{Q}_1\mathbf{A} = \begin{pmatrix} -2 & -5 \\ 0 & 0 \\ 0 & 1 \\ 0 & 2 \end{pmatrix}.$$

Теперь построим матрицу отражения $\mathbf{Q}_2$, которая зануляет элементы ниже главной диагонали во втором столбце матрицы $\mathbf{Q}_1\mathbf{A}$. Получаем

$$\mathbf{Q}_2 = \frac{1}{5}\begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & \sqrt{5} & 2\sqrt{5} \\ 0 & \sqrt{5} & 4 & -2 \\ 0 & 2\sqrt{5} & -2 & 1 \end{pmatrix}, \quad \tilde{\mathbf{A}} = \mathbf{Q}_2\mathbf{Q}_1\mathbf{A} = \begin{pmatrix} -2 & -5 \\ 0 & \sqrt{5} \\ 0 & 0 \\ 0 & 0 \end{pmatrix}.$$

Таким образом, матрица $\mathbf{Q} = \mathbf{Q}_2\mathbf{Q}_1$ приводит матрицу $\mathbf{A}$ к верхней треугольной матрице $\tilde{\mathbf{A}}$. Окончательно

$$\tilde{\mathbf{b}} = \mathbf{Q}\mathbf{b} = \mathbf{Q}_2\mathbf{Q}_1\mathbf{b} = \begin{pmatrix} -2,75 \\ 0,27\sqrt{5} \\ -0,02 - 0,01\sqrt{5} \\ 0,01 - 0,02\sqrt{5} \end{pmatrix}.$$

Поскольку $\|\mathbf{b} - \mathbf{A}\mathbf{c}\|_2^2 = \|\tilde{\mathbf{b}} - \tilde{\mathbf{A}}\mathbf{c}\|_2^2$, то решение задачи МНК получаем путем решения линейной системы с треугольной матрицей

$$\begin{pmatrix} -2 & -5 \\ 0 & \sqrt{5} \end{pmatrix}\begin{pmatrix} c_0 \\ c_1 \end{pmatrix} = \begin{pmatrix} -2,75 \\ 0,27\sqrt{5} \end{pmatrix}.$$

Отсюда $c_0^* = 0,70$ и $c_1^* = 0,27$. Отметим, что тот же самый результат мы получили выше, решая нормальную систему МНК. Вдобавок ко всему

$$\|\mathbf{b} - \mathbf{A}\mathbf{c}\|_2^2 = \|\tilde{\mathbf{b}} - \tilde{\mathbf{A}}\mathbf{c}\|_2^2 = (-0,02 - 0,01\sqrt{5})^2 + (0,01 - 0,02\sqrt{5})^2 = 0,003,$$

что также согласуется с ранее полученным результатом.

§ 5.13. Предобуславливание

Прежде, чем переходить к изложению итерационных методов решения систем линейных алгебраических уравнений, рассмотрим вопрос о способах подготовки таких систем к проведению итераций. Предположим, что мы решаем систему n уравнений с n неизвестными и невырожденной матрицей

$$\mathbf{A}\mathbf{x} = \mathbf{b}, \tag{5.27}$$

Для всякой невырожденной матрицы $\mathbf{M}$ порядка $n \times n$ система

$$\mathbf{M}^{-1}\mathbf{A}\mathbf{x} = \mathbf{M}^{-1}\mathbf{b} \tag{5.28}$$

будет иметь то же решение, что и система (1.27). Однако если мы будем решать систему (1.28) итерационно, то скорость сходимости будет зависеть от свойств матрицы $\mathbf{M}^{-1}\mathbf{A}$ вместо $\mathbf{A}$. Если *предобуславливатель* $\mathbf{M}$ выбран удачно, то система (1.28) может быть решена намного быстрее, чем система (1.28).

Для практического применения этой идеи необходимо иметь возможность эффективно проводить операции, связанные с произведением $\mathbf{M}^{-1}\mathbf{A}$. Нет необходимости вычислять матрицу $\mathbf{M}^{-1}$ в явном виде. Это может быть сделано путем решения системы

$$\mathbf{M}\mathbf{y} = \mathbf{c}. \tag{5.29}$$

Тривиальные варианты выбора предобуславливателя $\mathbf{M} = \mathbf{A}$ или $\mathbf{M} = \mathbf{E}$ не дают какого-либо выигрыша по сравнению с решением системы (1.27). Однако между этими двумя крайностями лежат предобуславливатели, которые могут быть получены путем быстрого решения системы (1.29) и будут достаточно близки к $\mathbf{A}$, чтобы обеспечить более быструю сходимость итераций для уравнения (1.28) по сравнению с уравнением (1.27).

Обычно считают, что матрица $\mathbf{M}$ близка к $\mathbf{A}$, если собственные числа матрицы $\mathbf{M}^{-1}\mathbf{A}$ близки к 1 и величина $\|\mathbf{M}^{-1}\mathbf{A} - \mathbf{E}\|_2$ достаточно мала. Во многих случаях этого достаточно, чтобы обеспечить быструю сходимость итераций для уравнения (1.28). Однако предобуславливатели, которые не удовлетворяют этим условиям, также часто работают хорошо. Как правило предобуславливатель $\mathbf{M}$ является хорошим, если матрица $\mathbf{M}^{-1}\mathbf{A}$ не слишком отличается от нормальной матрицы ($\mathbf{A}^T\mathbf{A} = \mathbf{A}\mathbf{A}^T$) и ее собственные значения сгруппированы, т. е. отношение максимального и минимального собственных чисел не слишком велико.

Уравнение (1.28) соответствует *левому предобуславливателю*. Если уравнение (1.27) преобразуется к виду $\mathbf{A}\mathbf{M}^{-1}\mathbf{y} = \mathbf{b}$, где $\mathbf{x} = \mathbf{M}^{-1}\mathbf{y}$, то используется *правый предобуславливатель*. На практике часто оба предобуславливателя применяются вместе.

Пусть, например, $\mathbf{A}$ – симметрическая положительно определенная матрица. При использовании предобуславливателя желательно сохранить это свойство. Возьмем положительно определенную матрицу $\mathbf{M} = \mathbf{C}\mathbf{C}^T$. Тогда равенство (1.28) эквивалентно уравнению

$$[\mathbf{C}^{-1}\mathbf{A}(\mathbf{C}^{-1})^T]\mathbf{C}^T\mathbf{x} = \mathbf{C}^{-1}\mathbf{b},$$

где матрица в квадратных скобках опять симметрическая и положительно определенная. Так как эта матрица подобна $\mathbf{M}^{-1}\mathbf{A}$, то они имеют одни и те же собственные числа.

Рассмотрим стационарный итерационный процесс вида

$$\mathbf{M}\mathbf{x}^{(k+1)} = \mathbf{M}\mathbf{x}^{(k)} - (\mathbf{A}\mathbf{x}^{(k)} - \mathbf{b}), \quad k = 0, 1, \ldots.$$

Матрицу $\mathbf{A}$ представим в виде суммы $\mathbf{A} = \mathbf{L} + \mathbf{D} + \mathbf{U}$, где $\mathbf{D}$ – диагональная матрица а $\mathbf{L}$ и $\mathbf{U}$ – нижняя и верхняя треугольные матрицы с нулями на диагонали. Приведем несколько нетривиальных вариантов выбора предобуславливателя $\mathbf{M}$:

1. метод простой итерации: $\mathbf{M} = \tau^{-1}\mathbf{E}$, $\tau > 0$,

$$\mathbf{x}^{(k+1)} = \mathbf{x}^{(k)} - \tau(\mathbf{A}\mathbf{x}^{(k)} - \mathbf{b});$$

2. метод Якоби: $\mathbf{M} = \mathbf{D}$,

$$\mathbf{x}^{(k+1)} = \mathbf{x}^{(k)} - \mathbf{D}^{-1}(\mathbf{A}\mathbf{x}^{(k)} - \mathbf{b}) \quad \text{или} \quad \mathbf{D}\mathbf{x}^{(k+1)} = \mathbf{b} - (\mathbf{L} + \mathbf{U})\mathbf{x}^{(k)};$$

3. метод релаксации: $\mathbf{M} = \mathbf{L} + \omega^{-1}\mathbf{D}$, $0 < \omega < 2$,

$$\mathbf{x}^{(k+1)} = \mathbf{x}^{(k)} - \omega(\omega\mathbf{L} + \mathbf{D})^{-1}(\mathbf{A}\mathbf{x}^{(k)} - \mathbf{b})$$

или

$$(\omega\mathbf{L} + \mathbf{D})\mathbf{x}^{(k+1)} = [(1 - \omega)\mathbf{D} - \omega\mathbf{U}]\mathbf{x}^{(k)} + \omega\mathbf{b}.$$

Далее эти и некоторые другие важные на практике итерационные процессы будут рассмотрены подробно. В приведенных случаях матрица $\mathbf{M}$ является диагональной или нижней треугольной. В приложениях важен также случай, когда в качестве $\mathbf{M}$ берется трехдиагональная матрица, образованная диагональю и первыми под- и наддиагоналями матрицы $\mathbf{A}$. Во всех трех случаях матрица $\mathbf{M}$ может быть легко обращена. Отметим, что предобуславливатель может не зависеть или зависеть от номера итерации, т. е. являться *стационарным* или *нестационарным*. Использование предобуславливателей особенно важно при численном решении дифференциальных уравнений с частными производными.

§ 5.14. Метод одновременных смещений Якоби

В системе (1.27) положим $\mathbf{A} = \mathbf{L} + \mathbf{D} + \mathbf{U}$, где $\mathbf{D}$ – диагональная матрица, а $\mathbf{L}$ и $\mathbf{U}$ – соответственно нижняя и верхняя треугольные матрицы с нулями на главной диагонали.

Допустим, что $a_{ii} \neq 0$, $i = 1, 2, \ldots, n$ и запишем уравнение (1.27) в виде

$$(\mathbf{L} + \mathbf{D} + \mathbf{U})\mathbf{x} = \mathbf{b}. \tag{5.30}$$

Систему (1.30) перепишем следующим образом:

$$\mathbf{D}\mathbf{x} = \mathbf{b} - (\mathbf{L} + \mathbf{U})\mathbf{x}.$$

Возьмем некоторый вектор начального приближения $\mathbf{x}^{(0)}$ и образуем итерации

$$\mathbf{D}\mathbf{x}^{(k+1)} = \mathbf{b} - (\mathbf{L} + \mathbf{U})\mathbf{x}^{(k)}, \quad k = 0, 1, \ldots . \tag{5.31}$$

В покомпонентной записи итерационный процесс (1.31) имеет вид:

$$x_i^{(k+1)} = \frac{1}{a_{ii}}\Big[b_i - \sum_{j=1}^{i-1} a_{ij}x_j^{(k)} - \sum_{j=i+1}^{n} a_{ij}x_j^{(k)}\Big], \quad i = 1, 2, \ldots, n,\ k = 0, 1, \ldots .$$

Пусть $\mathbf{x}^*$ – точное решение системы (1.30), а $\mathbf{w}^{(k)} = \mathbf{x}^{(k)} - \mathbf{x}^*$ – погрешность приближенного решения на k-й итерации. Перепишем равенство (1.31) в виде

$$\mathbf{D}(\mathbf{w}^{(k+1)} + \mathbf{x}^*) = \mathbf{b} - (\mathbf{L} + \mathbf{U})(\mathbf{w}^{(k)} + \mathbf{x}^*)$$

или, в силу соотношения (1.30),

$$\mathbf{D}\mathbf{w}^{(k+1)} = -(\mathbf{L} + \mathbf{U})\mathbf{w}^{(k)}.$$

Отсюда следует оценка

$$\|\mathbf{w}^{(k+1)}\| \leq \|\mathbf{C}\| \cdot \|\mathbf{w}^{(k)}\|, \quad \mathbf{C} = -\mathbf{D}^{-1}(\mathbf{L} + \mathbf{U})$$

и метод Якоби сходится, если для матрицы перехода $\mathbf{C}$ выполнено условие

$$\|\mathbf{C}\| < 1.$$

Итерационный процесс продолжается до тех пор, пока все $x_i^{(k+1)}$ и $x_i^{(k)}$ не станут достаточно близки. Критерий близости, например, можно задать в следующем виде:

$$\max_i |x_i^{(k+1)} - x_i^{(k)}| < \varepsilon.$$

При выполнении этого условия итерационный процесс следует остановить. Можно рассмотреть критерий, основанный на относительной ошибке

$$\max_i \left| \frac{x_i^{(k+1)} - x_i^{(k)}}{x_i^{(k)}} \right| < \varepsilon.$$

Покажем, что если $\mathbf{A}$ — матрица с диагональным преобладанием, то метод Якоби сходится. Воспользуемся следующим результатом.

Теорема 1.3 (Гершгорина). *Собственные числа λ комплексной квадратной матрицы $\mathbf{A}$ порядка n лежат в замкнутой области комплексной плоскости, являющейся объединением кругов:*

$$|a_{ii} - \lambda| \le \sum_{\substack{j=1 \\ j \ne i}}^{n} |a_{ij}|; \quad i = 1, 2, \ldots, n.$$

Доказательство. Пусть $\mathbf{A}$ – произвольная матрица порядка $n \times n$ с комплексными элементами и λ – некоторое её собственное число. Тогда матрица $\mathbf{A} - \lambda\mathbf{E}$ вырождена и существуют такие числа $x_1, x_2, \ldots, x_n$ с максимальным $|x_k|$, что

$$(a_{kk} - \lambda)x_k + \sum_{\substack{j=1 \\ j \ne k}}^{n} a_{kj}x_j = 0.$$

Но тогда

$$|a_{kk} - \lambda||x_k| \le \sum_{\substack{j=1 \\ j \ne k}}^{n} |a_{kj}||x_j| \le |x_k| \sum_{\substack{j=1 \\ j \ne k}}^{n} |a_{kj}|.$$

Сокращая на $|x_k|$, получаем

$$|a_{kk} - \lambda| \le \sum_{\substack{j=1 \\ j \ne k}}^{n} |a_{kj}|.$$

Аналогичное неравенство можно выписать для любого другого собственного числа матрицы $\mathbf{A}$. Каждое из этих соотношений определяет некоторый круг в комплексной λ-плоскости с центром в точке a_{ii} радиуса $\sum_{j=1, j\ne i}^{n} |a_{ij}|$. Поэтому все собственные числа матрицы $\mathbf{A}$ лежат в объединении этих кругов. Теорема доказана.

Если теперь $\mathbf{A}$ – матрица с диагональным преобладанием, то согласно теореме 1.3 для матрицы перехода $\mathbf{C}$ в методе Якоби имеем:

$$|\lambda_i| \le \frac{1}{|a_{ii}|} \sum_{\substack{j=1 \\ j \ne i}}^{n} |a_{ij}| < 1, \quad i = 1, 2, \ldots, n,$$

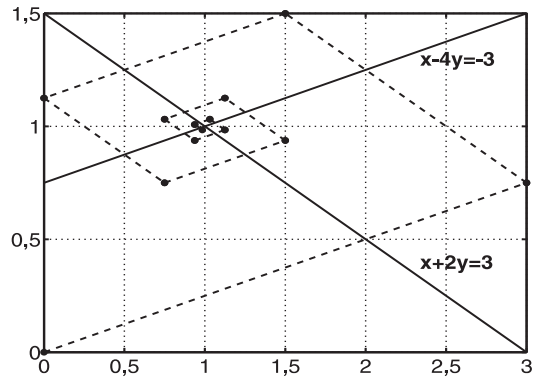


Рис. 5.2. Геометрическое представление сходимости метода Якоби при решении системы уравнений $x + 2y = 3$, $x - 4y = -3$

т. е. собственные числа матрицы перехода $\mathbf{C}$ удовлетворяют условию $|\lambda_i| < 1$ для всех i. Следовательно, $\|\mathbf{C}\|_2 < 1$ и метод Якоби сходится.

Таким образом, справедливо следующее утверждение.

Теорема 1.4. *Если* $\mathbf{A}$ *– матрица с диагональным преобладанием, то метод Якоби сходится.*

Отметим, что условия диагонального преобладания матрицы $\mathbf{A}$ являются достаточными условиями сходимости. Матрица может не обладать диагональным преобладанием, но метод Якоби все равно может сходиться.

Пример 1.13. Рассмотрим систему линейных уравнений:

$$\begin{cases} x + 2y = 3, \\ x - 4y = -3. \end{cases}$$

Здесь в первом уравнении нарушено условие диагонального преобладания. Во втором уравнении это условие выполняется. Тем не менее, полагая $(x_0, y_0) = (0, 0)$ и выбирая $\varepsilon = 0,01$, получаем сходимость метода Якоби за 15 итераций к точному решению системы $(x^*, y^*) = (1, 1)$. Рис. 1.2 иллюстрирует сходимость метода Якоби.

В данном примере матрица перехода $\mathbf{C}$ имеет вид:

$$\mathbf{C} = -\mathbf{D}^{-1}(\mathbf{L} + \mathbf{U}) = \begin{pmatrix} 0 & -2 \\ 1/4 & 0 \end{pmatrix}.$$

Для собственных чисел матрицы $\mathbf{C}$ получаем $|\lambda_{1,2}| < 1$, что и объясняет сходимость метода Якоби.

§ 5.15. Метод последовательных смещений Зейделя

Перепишем систему (1.30) в виде

$$(\mathbf{L} + \mathbf{D})\mathbf{x} = \mathbf{b} - \mathbf{U}\mathbf{x}$$

и образуем итерации:

$$(\mathbf{L}+\mathbf{D})\mathbf{x}^{(k+1)} = \mathbf{b} - \mathbf{U}\mathbf{x}^{(k)}, \quad k = 0, 1, \ldots. \tag{5.32}$$

Для погрешности получаем уравнения:

$$(\mathbf{L}+\mathbf{D})\mathbf{w}^{(k+1)} = -\mathbf{U}\mathbf{w}^{(k)}, \quad k = 0, 1, \ldots$$

или

$$\mathbf{w}^{(k+1)} = \mathbf{C}\mathbf{w}^{(k)}, \quad \mathbf{C} = -(\mathbf{L}+\mathbf{D})^{-1}\mathbf{U}.$$

Метод Зейделя сходится, если выполнено условие

$$\|\mathbf{C}\| < 1.$$

В покомпонентной записи итерационный процесс (1.31) имеет вид:

$$x_i^{(k+1)} = \frac{1}{a_{ii}}\Big[b_i - \sum_{j=1}^{i-1} a_{ij}x_j^{(k+1)} - \sum_{j=i+1}^{n} a_{ij}x_j^{(k)}\Big], \ i = 1, 2, \ldots, n, \ k = 1, 2, \ldots$$

Как следует из этой формулы, метод Зейделя характеризуется тем свойством, что на очередном шаге найденные компоненты вектора $\mathbf{x}^{(k+1)}$ сразу же используются в процессе итерации. Это приводит к ускорению сходимости по сравнению с методом Якоби. Отметим однако, что хотя метод Якоби сходится медленнее, но область его сходимости несколько шире, чем у метода Зейделя.

Теорема 1.5. *Если матрица* $\mathbf{A}$ *имеет диагональное преобладание, то метод Зейделя сходится.*

Доказательство. Пусть вектор $\mathbf{x}$ удовлетворяет характеристическому уравнению $\mathbf{C}\mathbf{x} = \lambda\mathbf{x}$ и имеет максимальную по модулю компоненту $|x_k|$. Тогда k-е уравнение системы $-\mathbf{U}\mathbf{x} = \lambda(\mathbf{L}+\mathbf{D})\mathbf{x}$ принимает вид:

$$-\sum_{j>k} a_{kj}x_j = \lambda\Big(a_{kk}x_k + \sum_{j<k} a_{kj}x_j\Big).$$

Отсюда при обозначении $\xi_j = x_j/|x_k|$ получаем:

$$|\lambda| \le \frac{\sum_{j>k}|a_{kj}||\xi_j|}{|a_{kk}| - \sum_{j<k}|a_{kj}||\xi_j|} \le \frac{\sum_{j>k}|a_{kj}|}{\sum_{j>k}|a_{kj}| + (|a_{kk}| - \sum_{j\ne k}|a_{kj}|)} < 1.$$

Таким образом, $\|\mathbf{C}\|_2 < 1$ и метод Зейделя сходится. Теорема доказана.

Отметим, что как и в методе Якоби условие диагонального преобладания матрицы $\mathbf{A}$ является только достаточным условием сходимости метода Зейделя. Обратимся опять к примеру 1.13, где матрица $\mathbf{A}$ не имеет диагонального преобладания. При $(x_0, y_0) = (0, 0)$ и $\varepsilon = 0,01$ метод Зейделя сходится за 10 итераций. Особенности его сходимости показаны на рис. 1.3. Согласно расчетным формулам:

$$\begin{cases} x_{k+1} = 3 - 2y_k, \\ y_{k+1} = (3 + x_{k+1})/4, \end{cases}$$

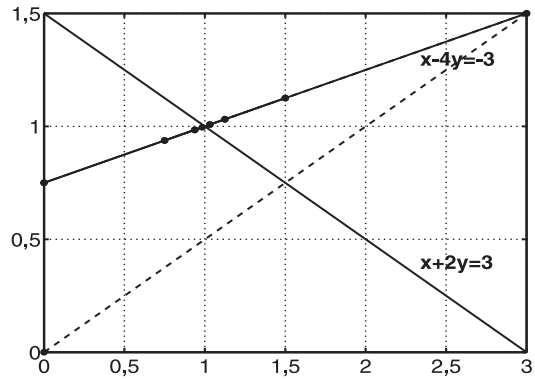


Рис. 5.3. Геометрическое представление сходимости метода Зейделя. Получаемые приближения располагаются на прямой $x - 4y = -3$

получаемые приближения располагаются на прямой $x - 4y = -3$. Матрица перехода $\mathbf{C} = -(\mathbf{L}+\mathbf{D})^{-1}\mathbf{U}$ имеет собственные числа $\lambda_1 = 0$, $\lambda_2 = -1/2$, и необходимые условия сходимости метода Зейделя выполнены.

В методе Зейделя для вектора приближенного решения

$$\mathbf{x}_i^{(k+1)} = (x_1^{(k+1)}, \ldots, x_{i-1}^{(k+1)}, x_i^{(k)}, \ldots, x_n^{(k)})^T$$

компоненты вектора невязки $\mathbf{r}_i^{(k+1)} = \mathbf{b} - \mathbf{A}\mathbf{x}_i^{(k+1)}$ вычисляются по формуле

$$r_{mi}^{(k+1)} = b_m - \sum_{j=1}^{i-1} a_{mj} x_j^{(k+1)} - \sum_{j=i}^{n} a_{mj} x_j^{(k)}, \quad m = 1, 2, \ldots, n.$$

Отсюда, в частности, следует, что итерации по Зейделю характеризуются выбором $x_i^{(k+1)}$ по правилу

$$x_i^{(k+1)} = x_i^{(k)} + \frac{r_{ii}^{(k+1)}}{a_{ii}}, \quad i = 1, 2, \ldots, n. \tag{5.33}$$

§ 5.16. Метод верхней релаксации

Модифицируем формулу (1.33), положив

$$x_i^{(k+1)} = x_i^{(k)} + \omega \frac{r_{ii}^{(k+1)}}{a_{ii}}, \quad i = 1, 2, \ldots, n, \tag{5.34}$$

где параметр ω удовлетворяет ограничениям $0 < \omega < 2$. При $\omega = 1$ получаем метод Зейделя. При $0 < \omega < 1$ формула (1.34) дает *метод нижней релаксации*. Можно показать, что область сходимости этого метода шире, чем у метода Зейделя. При $1 < \omega < 2$ получаем *метод верхней релаксации*. Можно показать, что этот метод сходится быстрее, чем метод Зейделя. При $\omega \leq 0$ или $\omega \geq 2$ метод (1.34) расходится.

Формулу (1.34) можно переписать в виде:

$$x_i^{(k+1)} = (1-\omega)x_i^{(k)} + \frac{\omega}{a_{ii}}\Big[b_i - \sum_{j=1}^{i-1} a_{ij}x_j^{(k+1)} - \sum_{j=i+1}^{n} a_{ij}x_j^{(k)}\Big], \quad i = 1, 2, \ldots, n$$

или в матричной форме:

$$(\omega\mathbf{L} + \mathbf{D})\mathbf{x}^{(k+1)} = [(1-\omega)\mathbf{D} - \omega\mathbf{U}]\mathbf{x}^{(k)} + \omega\mathbf{b}.$$

Отсюда для погрешности $\mathbf{w}^{(k)} = \mathbf{x}^{(k)} - \mathbf{x}^*$ получаем

$$(\omega\mathbf{L} + \mathbf{D})\mathbf{w}^{(k+1)} = [(1-\omega)\mathbf{D} - \omega\mathbf{U}]\mathbf{w}^{(k)}.$$

Это позволяет выписать оценку

$$\|\mathbf{w}^{(k+1)}\| \le \|\mathbf{C}\|\,\|\mathbf{w}^{(k)}\|, \quad \mathbf{C} = (\omega\mathbf{L} + \mathbf{D})^{-1}[(1-\omega)\mathbf{D} - \omega\mathbf{U}].$$

Метод верхней релаксации сходится, если и только если выполнено условие

$$\|\mathbf{C}\| < 1.$$

Приведем без доказательства следующее утверждение.

Теорема 1.6. *Для того, чтобы метод релаксации* (1.34) *для системы* (1.27) *с вещественной симметрической матрицей* $\mathbf{A}$*, имеющей положительные диагональные элементы, сходился, необходимо и достаточно, чтобы матрица* $\mathbf{A}$ *была положительной, т. е. все ее собственные числа были бы положительны.*

Пусть $\lambda_{\max}$ – максимальное собственное число матрицы $\mathbf{C} = -\mathbf{D}^{-1}(\mathbf{L} + \mathbf{U})$ в методе Якоби. Согласно теории, развитой Янгом в [4], в случае вещественной симметрической положительно определенной матрицы $\mathbf{A}$ оптимальное значение параметра верхней релаксации ω вычисляется по формуле

$$\omega = \frac{2}{1 + \sqrt{1 - \lambda_{max}^2(\mathbf{C})}}. \tag{5.35}$$

Отметим, что если размерность матрицы $\mathbf{A}$ невелика, то ω близко к единице. При увеличении числа уравнений ω растет и в пределе стремится к двум.

Пример 1.14. Рассмотрим систему линейных уравнений $\mathbf{Ax} = \mathbf{b}$, где матрица $\mathbf{A}$ имеет вид:

$$\mathbf{A} = \begin{pmatrix} 2 & -1 \\ -1 & 2 \end{pmatrix}.$$

Матрицу $\mathbf{A}$ разобьем на три части, положив $\mathbf{A} = \mathbf{L} + \mathbf{D} + \mathbf{U}$, где

$$\mathbf{L} = \begin{pmatrix} 0 & 0 \\ -1 & 0 \end{pmatrix}, \quad \mathbf{D} = \begin{pmatrix} 2 & 0 \\ 0 & 2 \end{pmatrix}, \quad \mathbf{U} = \begin{pmatrix} 0 & -1 \\ 0 & 0 \end{pmatrix}.$$

Для метода Якоби матрица перехода $\mathbf{C} = \mathbf{D}^{-1}(\mathbf{L} + \mathbf{U})$ имеет вид:

$$\mathbf{C} = \begin{pmatrix} 1/2 & 0 \\ 0 & 1/2 \end{pmatrix} \cdot \begin{pmatrix} 0 & -1 \\ -1 & 0 \end{pmatrix} = \begin{pmatrix} 0 & -1/2 \\ -1/2 & 0 \end{pmatrix}.$$

Решая характеристическое уравнение $\det(\mathbf{C} - \lambda\mathbf{E}) = 0$, имеем $\lambda_{1,2} = \pm\frac{1}{2}$. Таким образом, $\|C\|_2 = \max_{i=1,2} |\lambda_i| = 1/2$ и, следовательно, на каждом шаге итерационного процесса Якоби погрешность убывает в два раза.

Для метода Зейделя матрица перехода $\mathbf{C} = (\mathbf{L} + \mathbf{D})^{-1}\mathbf{U}$ имеет вид:

$$\mathbf{C} = \begin{pmatrix} 1/2 & 0 \\ 1/4 & 1/2 \end{pmatrix} \cdot \begin{pmatrix} 0 & -1 \\ 0 & 0 \end{pmatrix} = \begin{pmatrix} 0 & -1/2 \\ 0 & 1/4 \end{pmatrix}.$$

Характеристическое уравнение $\det(\mathbf{C} - \lambda\mathbf{E}) = 0$, имеет корни $\lambda_1 = -1/4$, $\lambda_2 = 0$. Таким образом, $\|C\|_2 = \max_{i=1,2} |\lambda_i| = 1/4$ и, следовательно, на каждом шаге итерационного процесса Зейделя погрешность убывает в четыре раза.

Матрица перехода $\mathbf{C} = (\omega\mathbf{L}+\mathbf{D})^{-1}[(1-\omega)\mathbf{D}-\omega\mathbf{U}]$ в методе верхней релаксации имеет вид:

$$\mathbf{C} = \begin{pmatrix} 1/2 & 0 \\ \omega/4 & 1/2 \end{pmatrix} \cdot \begin{pmatrix} 2(1-\omega) & \omega \\ 0 & 2(1-\omega) \end{pmatrix} = \begin{pmatrix} 1-\omega & \omega/2 \\ \omega(1-\omega)/2 & 1-\omega+\omega^2/4 \end{pmatrix}.$$

Характеристическое уравнение $\det(\mathbf{C} - \lambda\mathbf{E}) = 0$ дает нам равенство

$$\lambda^2 - \lambda\Big[2(1-\omega) + \frac{\omega^2}{4}\Big] + (1-\omega)^2 = 0.$$

Согласно теореме Виета для корней этого уравнения λ_1 и λ_2 (т. е. для следа и детерминанта матрицы $\mathbf{C} - \lambda\mathbf{E}$) имеем:

$$\begin{aligned} \lambda_1 + \lambda_2 &= 2(1-\omega) + \frac{\omega^2}{4}, \\ \lambda_1\lambda_2 &= (1-\omega)^2. \end{aligned}$$

Если предположить, что оптимальное ω, удовлетворяющее условию $1 < \omega < 2$, отвечает случаю кратных корней $\lambda_1 = \lambda_2 = \omega - 1$, то для нахождения ω получаем квадратное уравнение

$$\frac{\omega^2}{4} - 4\omega + 4 = 0,$$

откуда имеем

$$\omega = 4(2 - \sqrt{3}) \approx 1,0718.$$

Отметим, что этот же результат может быть получен, используя формулу (1.35).

Таким образом, $\|C\|_2 = \max_{i=1,2} |\lambda_i| = \omega - 1 \approx \frac{1}{14}$ и, следовательно, при оптимальном выборе параметра релаксации ω на каждом шаге метода верхней релаксации погрешность убывает примерно в 14 раз.

Пример 1.15. Пусть матрица системы линейных уравнений (1.27) имеет вид

$$\mathbf{A} = \begin{pmatrix} p & 1 & 0 & 0 \\ 1 & p & 1 & 0 \\ 0 & 1 & p & 1 \\ 0 & 0 & 1 & p \end{pmatrix}, \quad |p| \geq 2. \tag{5.36}$$

В методе Якоби для матрицы перехода $\mathbf{C} = -\mathbf{D}^{-1}(\mathbf{L} + \mathbf{U})$ получаем

$$-\begin{pmatrix} p^{-1} & 0 & 0 & 0 \\ 0 & p^{-1} & 0 & 0 \\ 0 & 0 & p^{-1} & 0 \\ 0 & 0 & 0 & p^{-1} \end{pmatrix} \begin{pmatrix} 0 & 1 & 0 & 0 \\ 1 & 0 & 1 & 0 \\ 0 & 1 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{pmatrix} = \begin{pmatrix} 0 & -p^{-1} & 0 & 0 \\ -p^{-1} & 0 & -p^{-1} & 0 \\ 0 & -p^{-1} & 0 & -p^{-1} \\ 0 & 0 & -p^{-1} & 0 \end{pmatrix}.$$

Так как здесь характеристическое уравнение

$$\det(\mathbf{C} - \lambda E) = \lambda^4 - \frac{3}{p^2}\lambda^2 + \frac{1}{p^4} = 0$$

имеет корни

$$\lambda^2_{1,2} = \frac{3 \pm \sqrt{5}}{2p^2} \quad \text{и} \quad \lambda^2_{\max} = \frac{3 + \sqrt{5}}{2p^2},$$

то по формуле Янга (1.35) можно легко найти ω.

При $p = 2$ для метода Якоби $\|\mathbf{C}\|_2 = \lambda_{\max} \approx 0,8090$, а для метода верхней релаксации $\|\mathbf{C}\|_2 = \omega - 1 = 0{,}2596$, т. е. имеем ускорение примерно в 3 раза.

При $p = 4$ для метода Якоби $\|\mathbf{C}\|_2 = \lambda_{\max} \approx 0{,}4045$ а для метода верхней релаксации $\|\mathbf{C}\|_2 = \omega - 1 = 0{,}0446$, т. е. имеем ускорение примерно в 9 раз.

§ 5.17. Метод простой итерации

Здесь для решения системы (1.27) итерации образуются по правилу

$$\mathbf{x}^{(k+1)} = \mathbf{x}^{(k)} - \tau(\mathbf{A}\mathbf{x}^{(k)} - \mathbf{b}); \quad k = 0, 1, \ldots.$$

Это уравнение, переписанное относительно вектора ошибки $\mathbf{w}^{(k)} = \mathbf{x}^{(k)} - \mathbf{x}^*$, принимает вид:

$$\mathbf{w}^{(k+1)} = \mathbf{C}\mathbf{w}^{(k)}, \quad \mathbf{C} = \mathbf{E} - \tau\mathbf{A}. \tag{5.37}$$

Отсюда следует оценка

$$\|\mathbf{w}^{(k+1)}\| \leq |\mathbf{C}|\,\|\mathbf{w}^{(k)}\|,$$

и метод простой итерации сходится, если

$$\|\mathbf{C}\| < 1.$$

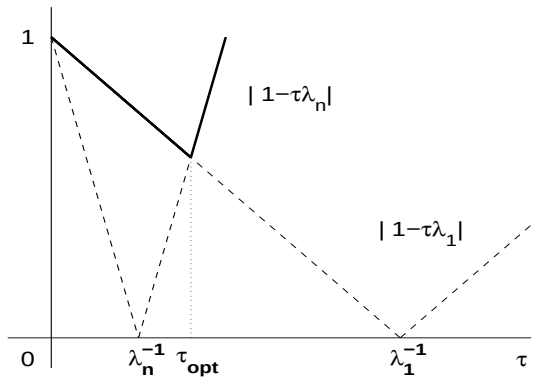


Рис. 5.4. Оптимизация итерационного параметра в методе простой итерации

Пусть $\mathbf{A}$ – симметрическая положительно определенная матрица. Тогда она имеет систему положительных собственных чисел $0 < \lambda_1 < \lambda_2 < \ldots < \lambda_n$ и систему ортогональных собственных векторов $\mathbf{u}_1, \ldots, \mathbf{u}_n$. Вектор ошибки приближения $\mathbf{w}^{(k)}$ можно записать в виде разложения по базису из собственных векторов:

$$\mathbf{w}^{(k)} = \sum_{i=1}^{n} c_i \mathbf{u}_i.$$

Подставляя это представление в уравнение ошибки (1.37), находим

$$\mathbf{w}^{(k+1)} = (\mathbf{E} - \tau\mathbf{A}) \sum_{i=1}^{n} c_i \mathbf{u}_i = \sum_{i=1}^{n} (1 - \tau\lambda_i) c_i \mathbf{u}_i.$$

Отсюда получаем оценку

$$\|\mathbf{w}^{(k+1)}\| \leq \max_i |1 - \tau\lambda_i| \cdot \|\mathbf{w}^{(k)}\|.$$

По предположению все собственные числа λ_i положительны и монотонно возрастают вместе с i. Поэтому

$$\min_{\tau} \max_{\lambda_1 \leq \lambda \leq \lambda_n} |1 - \tau\lambda| = \min_{\tau} \max[|1 - \tau\lambda_1|, |1 - \tau\lambda_n|],$$

и оптимальное значение итерационного параметра τ вычисляется по правилу (рис. 1.4)

$$1 - \tau\lambda_1 = -(1 - \tau\lambda_n),$$

откуда

$$\tau_{opt} = \frac{2}{\lambda_1 + \lambda_n}.$$

Пример 1.16. Снова рассмотрим матрицу (1.36). Характеристическое уравнение для этой матрицы

$$(p - \lambda)^4 - 3(p - \lambda)^2 + 1 = 0$$

имеет корни

$$\lambda_{1,4} = p \pm \sqrt{\frac{3+\sqrt{5}}{2}}, \quad \lambda_{2,3} = p \pm \sqrt{\frac{3-\sqrt{5}}{2}}.$$

Поэтому в данном случае

$$\tau_{opt} = \frac{2}{\lambda_1 + \lambda_4} = p^{-1}.$$

Для ускорения сходимости метода простой итерации бывает полезно предварительно подготовить систему (1.27), переписав ее, например, в виде

$$\mathbf{MAx} = \mathbf{Mb},$$

где $\mathbf{M}$ – некоторая предобуславливающая невырожденная матрица такая, что $\mathbf{MA}$ близка к единичной матрице. В этом случае рассматривается итерационный процесс:

$$\mathbf{x}^{(k+1)} = \mathbf{x}^{(k)} - \mathbf{M}(\mathbf{Ax}^{(k)} - \mathbf{b}), \quad k = 0, 1, \ldots.$$

Если, например, $\mathbf{M} = \mathbf{D}^{-1}$, то это дает нам метод Якоби (см. § 1.12).

Можно показать, что метод простой итерации и метод Зейделя имеют различные области сходимости, которые частично перекрываются [29].

Пример 1.17. Покажем, что для линейной системы $\mathbf{Ax} = \mathbf{b}$, где

$$\mathbf{A} = \begin{pmatrix} 2 & 0{,}3 & 0{,}5 \\ 0{,}1 & 3 & 0{,}4 \\ 0{,}1 & 0{,}1 & 4{,}8 \end{pmatrix}$$

метод простой итерации сходится при $0 < \tau < 0{,}4$.

Как было показано выше, метод простой итерации сходится, если для матрицы перехода $\mathbf{C} = \mathbf{E} - \tau\mathbf{A}$ выполняется неравенство $\|\mathbf{C}\| < 1$. В нашем случае

$$\mathbf{C} = \begin{pmatrix} 1-2\tau & -0{,}3\tau & -0{,}5\tau \\ -0{,}1\tau & 1-3\tau & -0{,}4\tau \\ -0{,}1\tau & -0{,}1\tau & 1-4{,}8\tau \end{pmatrix}.$$

Для оценки собственных чисел матрицы $\mathbf{C}$ воспользуемся теоремой 1.3 (Гершгорина), согласно которой можно записать:

$$\begin{aligned} |1-2\tau-\lambda| &\le 0{,}8\tau, \\ |1-3\tau-\lambda| &\le 0{,}5\tau, \\ |1-4{,}8\tau-\lambda| &\le 0{,}2\tau \end{aligned}$$

или, с учетом условия $|\lambda| < 1$, получаем:

$$-1 < 1 - 2{,}8\tau \le \lambda \le 1 - 1{,}2\tau < 1,$$

$$-1 < 1 - 3{,}5\tau \le \lambda \le 1 - 2{,}5\tau < 1,$$
$$-1 < 1 - 5{,}0\tau \le \lambda \le 1 - 4{,}6\tau < 1.$$

Отсюда следуют неравенства:

$$0 < \tau < \frac{1}{1{,}4} \approx 0{,}714,$$
$$0 < \tau < \frac{2}{3{,}5} \approx 0{,}571,$$
$$0 < \tau < \frac{2}{5} = 0{,}4.$$

Таким образом, при $0 < \tau < 0{,}4$ все собственные числа матрицы перехода $\mathbf{C}$ по модулю меньше единицы. Так как эта матрица не является симметрической, то отсюда еще нельзя сделать вывод о сходимости метода простой итерации. Однако при $0 < \tau < 0{,}4$ для сумм модулей элементов по строкам матрицы $\mathbf{C}$ имеем

$$|1 - 2\tau| + 0{,}3\tau + 0{,}5\tau < 1,$$
$$0{,}1\tau + |1 - 3\tau| + 0{,}4\tau < 1,$$
$$0{,}1\tau + 0{,}1\tau + |1 - 4{,}8\tau| < 1.$$

Следовательно, $\|\mathbf{C}\|_\infty < 1$ и метод простой итерации сходится.

§ 5.18. Метод Ричардсона

Этот метод является обобщением метода простой итерации. Пусть $\mathbf{A}$ – симметрическая положительно определенная матрица и известны собственные значения матрицы $\mathbf{A}$: $0 < \lambda_1 < \lambda_2 < \ldots < \lambda_n$. Рассмотрим циклы из l шагов

$$\mathbf{x}^{(k+1)} = \mathbf{x}^{(k)} - \tau_k(\mathbf{A}\mathbf{x}^{(k)} - \mathbf{b}); \quad k = 0, 1, \ldots, l-1.$$

В результате на l-м шаге уравнение ошибки имеет вид:

$$\mathbf{w}^{(l)} = \mathbf{w}^{(l-1)} - \tau_{l-1}\mathbf{A}\mathbf{w}^{(l-1)} = (\mathbf{E} - \tau_{l-1}\mathbf{A}) \ldots (\mathbf{E} - \tau_0\mathbf{A})\mathbf{w}^{(0)}.$$

Отсюда следует оценка

$$\|\mathbf{w}^{(l)}\| \le \max_{\lambda_1 \le \lambda \le \lambda_n} |Q_l(\lambda)| \, \|\mathbf{w}^{(0)}\|, \quad Q_l(\lambda) = (1 - \tau_{l-1}\lambda) \ldots (1 - \tau_0\lambda),$$

и задача об оптимальном выборе итерационных параметров $\tau_0, \ldots, \tau_{l-1}$ сводится к нахождению минимума величины

$$\min_{\tau_0, \ldots, \tau_{l-1}} \max_{\lambda_1 \le \lambda \le \lambda_n} |Q_l(\lambda)|. \tag{5.38}$$

В задаче (1.38) среди всех многочленов l-й степени и таких, что $Q_l(0) = 1$, требуется найти многочлен, наименее уклоняющийся от нуля на отрезке $[\lambda_1, \lambda_n]$. Решением данной задачи является многочлен А. А. Маркова, тесно связанный с многочленом П. Л. Чебышева 1-го рода. В этом случае оптимальный набор итерационных параметров имеет вид:

$$\tau_i = 2\Big[\lambda_1 + \lambda_n + (\lambda_n - \lambda_1)\cos\frac{\pi}{l}\Big(i + \frac{1}{2}\Big)\Big]^{-1}; \quad i = 0, 1, \ldots, l-1.$$

При практическом применении метода Ричардсона требуется специальное упорядочивание итерационных параметров τ_i. Это вызвано тем, что при расчете на ЭВМ с конечным числом разрядов не любая фиксация $\tau_{i_0}, \tau_{i_1}, \ldots, \tau_{i_{l-1}}$ приводит к правильному результату. Возможен авост внутри цикла или сильное искажение решения [16].

§ 5.19. Метод наискорейшего градиентного спуска

Этот метод также является обобщением метода простой итерации. Однако здесь выбор оптимальных итерационных параметров основывается на использовании так называемого вариационного подхода и не требует знания наибольшего и наименьшего собственных чисел матрицы $\mathbf{A}$.

Нам потребуется понятие линейного функционала J, который определим как функцию $J : \mathbb{R}^n \to \mathbb{R}$ такую, что для любых векторов $\mathbf{x}, \mathbf{y} \in \mathbb{R}^n$ и постоянных α, β имеем

$$J(\alpha\mathbf{x} + \beta\mathbf{y}) = \alpha J(\mathbf{x}) + \beta J(\mathbf{y}).$$

Примером линейного функционала является скалярное произведение векторов в $\mathbb{R}^n$

$$(\mathbf{x}, \mathbf{y}) = \sum_{i=1}^{n} x_i y_i.$$

Известно, что всякий линейный функционал J может быть представлен в виде скалярного произведения (теорема Рисса-Фишера)

$$J(\mathbf{x}) = (\mathbf{x}, \mathbf{x}_0),$$

где вектор $\mathbf{x}_0 \in \mathbb{R}^n$ однозначно определяется функционалом J.

Будем считать, что матрица системы $\mathbf{Ax} = \mathbf{b}$ симметрична и положительно определена. Для таких матриц удобной мерой погрешности является функционал ошибки

$$J(\mathbf{x}) = (\mathbf{A}(\mathbf{x} - \mathbf{x}^*), \mathbf{x} - \mathbf{x}^*),$$

где $\mathbf{x}^*$ – точное решение системы. В силу положительной определенности $\mathbf{A}$ всегда $J(\mathbf{x}) \geq 0$, причем $J(\mathbf{x}) = 0$ только при $\mathbf{x} = \mathbf{x}^*$.

Очевидно, что

$$J(\mathbf{x}) = (\mathbf{A}\mathbf{x}, \mathbf{x}) - 2(\mathbf{b}, \mathbf{x}) + (\mathbf{b}, \mathbf{x}^*).$$

Этот функционал однако не может быть вычислен, если не известно $\mathbf{x}^*$. Поэтому разумно рассмотреть функционал

$$J_0(\mathbf{x}) = (\mathbf{A}\mathbf{x}, \mathbf{x}) - 2(\mathbf{b}, \mathbf{x}),$$

который отличается от J постоянным слагаемым $(\mathbf{b}, \mathbf{x}^*)$ и позволяет контролировать убывание функционала ошибки J.

еорема 1.7. *Пусть матрица* $\mathbf{A}$ *симметрична и положительно определена. Вектор* $\mathbf{x}^*$ *является решением системы* $\mathbf{A}\mathbf{x} = \mathbf{b}$*, если и только если* $\mathbf{x}^*$ *доставляет минимум функционалу* J_0*.*

Доказательство. Пусть $\mathbf{x}^*$ – решение системы (1.27). Рассмотрим произвольный вектор $\mathbf{x} \in \mathbb{R}$, который запишем в виде $\mathbf{x} = \mathbf{x}^* + \mathbf{z}$. Тогда

$$\begin{aligned} J_0(\mathbf{x}^* + \mathbf{z}) &= (\mathbf{A}(\mathbf{x}^* + \mathbf{z}), \mathbf{x}^* + \mathbf{z}) - 2(\mathbf{b}, \mathbf{x}^* + \mathbf{z}) = \\ &= (\mathbf{A}\mathbf{x}^*, \mathbf{x}^*) - 2(\mathbf{b}, \mathbf{x}^*) + (\mathbf{A}\mathbf{z}, \mathbf{z}) = J_0(\mathbf{x}^*) + (\mathbf{A}\mathbf{z}, \mathbf{z}) > J_0(\mathbf{x}^*). \end{aligned}$$

т. е. функционал J_0 достигает минимума на решении системы (1.27).

Пусть теперь вектор $\mathbf{x}^* \in \mathbb{R}^n$ доставляет минимум функционалу J_0. Рассмотрим вектор $\mathbf{x}^* + \tau\mathbf{y}$, где τ – вещественный параметр, а $\mathbf{y}$ – произвольный вектор из $\mathbb{R}^n$. Условия минимума функционала J_0 в точке $\mathbf{x}^*$ имеют вид:

$$\frac{d}{d\tau} J_0(\mathbf{x}^* + \tau\mathbf{y})\Big|_{\tau=0} = 0, \quad \frac{d^2}{d\tau^2} J_0(\mathbf{x}^* + \tau\mathbf{y})\Big|_{\tau=0} > 0.$$

Поскольку

$$\frac{d}{d\tau} J_0(\mathbf{x}^* + \tau\mathbf{y}) = 2(\mathbf{A}\mathbf{x}^* - \mathbf{b}, \mathbf{y}) + 2\tau(\mathbf{A}\mathbf{y}, \mathbf{y}), \quad \frac{d^2}{d\tau^2} J_0(\mathbf{x}^* + \tau\mathbf{y}) = 2(\mathbf{A}\mathbf{y}, \mathbf{y}),$$

то условия минимума функционала J_0 эквивалентны соотношениям

$$(\mathbf{A}\mathbf{x}^* - \mathbf{b}, \mathbf{y}) = 0 \quad \text{для всех} \quad \mathbf{y} \in \mathbb{R}^n, \quad (\mathbf{A}\mathbf{y}, \mathbf{y}) > 0.$$

В силу произвольности вектора $\mathbf{y}$ первое из этих соотношений означает ортогональность вектора $\mathbf{A}\mathbf{x}^* - \mathbf{b}$ любому вектору из $\mathbb{R}^n$ и, следовательно, $\mathbf{A}\mathbf{x}^* - \mathbf{b} = \mathbf{0}$. Таким образом, вектор $\mathbf{x}^*$ является решением системы (1.27). Выполнение второго соотношения очевидно в силу положительной определенности матрицы $\mathbf{A}$. Теорема доказана.

Распространенным методом минимизации функций многих переменных является *метод градиентного спуска*. Последующее приближение получается в нем из предыдущего смещением в направлении, противоположном градиенту функции F, т. е.

$$\mathbf{x}^{(k+1)} = \mathbf{x}^{(k)} - \alpha_k \operatorname{grad} F(\mathbf{x}^{(k)}),$$

где α_k – итерационный параметр.

Для функции $J_0(\mathbf{x}) = (\mathbf{Ax}, \mathbf{x}) - 2(\mathbf{b}, \mathbf{x})$, соответствующей системе (1.27), имеем

$$\operatorname{grad} J_0(\mathbf{x}) = 2(\mathbf{Ax} - \mathbf{b})$$

и расчетные формулы метода наискорейшего спуска получают вид

$$\mathbf{x}^{(k+1)} = \mathbf{x}^{(k)} - \tau_k(\mathbf{Ax}^{(k)} - \mathbf{b}), \quad k = 0, 1, \ldots, \tag{5.39}$$

где $\tau_k = 2\alpha_k$. Таким образом, мы приходим к формуле метода простой итерации с переменным параметром τ_k.

Оптимальную последовательность итерационных параметров будем строить, исходя из условия минимизации функционала J_0 вдоль выбранного направления спуска. Поскольку

$$\begin{aligned} J_0(\mathbf{x}^{(k+1)}) &= (\mathbf{Ax}^{(k+1)}, \mathbf{x}^{(k+1)}) - 2(\mathbf{b}, \mathbf{x}^{(k+1)}) = \\ &= J(\mathbf{x}^{(k)}) - 2\tau_k(\mathbf{r}^{(k)}, \mathbf{r}^{(k)}) + \tau_k^2(\mathbf{Ar}^{(k)}, \mathbf{r}^{(k)}), \quad \mathbf{r}^{(k)} = \mathbf{b} - \mathbf{Ax}^{(k)}, \end{aligned}$$

то минимум функционала J_0 вдоль направления $\operatorname{grad} F(\mathbf{x}^{(k)})$ имеем при

$$\tau_k = \frac{(\mathbf{r}^{(k)}, \mathbf{r}^{(k)})}{(\mathbf{Ar}^{(k)}, \mathbf{r}^{(k)})}. \tag{5.40}$$

Итерационный процесс (1.39), (1.40) принято называть *методом наискорейшего градиентного спуска* или просто *методом наискорейшего спуска.* На рис. 1.5 изображены последовательные приближения метода наискорейшего спуска и линии уровня функции J_0.

В целях экономии вычислений, чтобы избежать двух трудоемких операций умножения матрицы на вектор, целесообразно переписать формулу (1.39) в виде

$$\mathbf{x}^{(k+1)} = \mathbf{x}^{(k)} + \tau_k\mathbf{r}^{(k)}, \quad k = 0, 1, \ldots. \tag{5.41}$$

Умножая теперь это равенство на $\mathbf{A}$ и вычитая $\mathbf{b}$, получаем

$$\mathbf{r}^{(k+1)} = \mathbf{r}^{(k)} - \tau_k\mathbf{Ar}^{(k)}, \quad k = 0, 1, \ldots. \tag{5.42}$$

Итерационный процесс (1.40)–(1.42) требует на итерацию только одного умножения матрицы на вектор. На каждом шаге запоминаются $\mathbf{x}^{(k)}$, $\mathbf{r}^{(k)}$ и вычисляются $\mathbf{Ar}^{(k)}$, τ_k, $\mathbf{x}^{(k+1)}$, $\mathbf{r}^{(k+1)}$.

Если известны минимальное и максимальное собственные числа симметрической положительно определенной матрицы $\mathbf{A}$, то можно показать [1], что для метода наискорейшего спуска справедлива оценка

$$J_0(\mathbf{x}^{(k+1)}) \le \left(\frac{\lambda_n - \lambda_1}{\lambda_n + \lambda_1}\right)^2 J_0(\mathbf{x}^{(k)}).$$

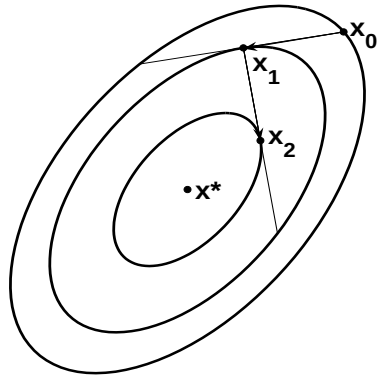


Рис. 5.5. Геометрическое представление сходимости метода наискорейшего спуска

§ 5.20. Регуляризация

Если система $\mathbf{Ax} = \mathbf{b}$ плохо обусловлена, то это значит, что малые погрешности коэффициентов матрицы и правой части или погрешности округления при счете могут сильно исказить решение. Тогда систему надо *регуляризовать*, т. е. изменить таким образом, чтобы малые изменения входных данных приводили к малому изменению решения.

Задача об отыскании минимума функционала $J(\mathbf{x}) = (\mathbf{Ax} - \mathbf{b}, \mathbf{Ax} - \mathbf{b})$ равносильна решению системы $\mathbf{Ax} = \mathbf{b}$. Рассмотрим связанную с ней задачу об отыскании минимума функционала

$$J_\alpha(\mathbf{x}) = J(\mathbf{x}) + \alpha(\mathbf{x} - \mathbf{x}_0, \mathbf{x} - \mathbf{x}_0), \quad \alpha > 0, \tag{5.43}$$

где $\mathbf{x}_0$ – заданный вектор. Скалярное произведение $(\mathbf{x} - \mathbf{x}_0, \mathbf{x} - \mathbf{x}_0)$ вводится здесь для того, чтобы регулировать величину отклонения решения $\mathbf{x}$ от вектора $\mathbf{x}_0$ и, вообще говоря, должно быть здесь минимально, поскольку решение должно как можно меньше отклоняться от заданного вектора $\mathbf{x}_0$.

Ясно, что функционал (1.43) можно переписать в эквивалентной форме:

$$\begin{aligned} J_\alpha(\mathbf{x}) \;\; = \;\; & (\mathbf{x}, \mathbf{A}^T\mathbf{Ax}) - 2(\mathbf{x}, \mathbf{A}^T\mathbf{b}) + (\mathbf{b}, \mathbf{b}) + \\ & + \alpha[(\mathbf{x}, \mathbf{x}) - 2(\mathbf{x}, \mathbf{x}_0) + (\mathbf{x}_0, \mathbf{x}_0)]. \end{aligned}$$

Необходимое условие минимума этого функционала $\operatorname{grad} J_\alpha(\mathbf{x}) = \mathbf{0}$ дает уравнение

$$(\mathbf{A}^T\mathbf{A} + \alpha\mathbf{E})\mathbf{x} = \mathbf{A}^T\mathbf{b} + \alpha\mathbf{x}_0 \tag{5.44}$$

где $\mathbf{E}$ – единичная матрица. Эта система будет хорошо обусловленной и к тому же имеет симметрическую матрицу, которую можно сделать даже положительно определенной за счет слагаемого $\alpha\mathbf{E}$. Решая эту систему, найдем регуляризованное решение $\mathbf{x}_\alpha$, зависящее от параметра α.

Каков же все-таки механизм регуляризации? За счет чего получающаяся система более устойчива к ошибкам счета? Нижняя граница спектра матрицы $\mathbf{A}^T\mathbf{A}+\alpha\mathbf{E}$ – как минимум α и поэтому число обусловленности $\sigma_{max}/\sigma_{min}$ для $\mathbf{A}^T\mathbf{A}+\alpha\mathbf{E}$ сильно падает в сравнении с таковым для $\mathbf{A}^T\mathbf{A}$. При этом σ_{max} почти не меняется.

Если $\alpha = 0$, то линейная система (1.44) переходит в плохообусловленную систему $\mathbf{A}^T\mathbf{A}\mathbf{x} = \mathbf{A}^T\mathbf{b}$. Если α велико, то система (1.44) будет хорошо обусловлена, но решение $\mathbf{x}_\alpha$ не будет близко к искомому решению. Оптимальным будет наименьшее значение α, при котором обусловленность системы (1.44) еще удовлетворительна.

Для нахождения оптимального α вычисляют невязку $\mathbf{r}_\alpha = \mathbf{b} - \mathbf{A}\mathbf{x}_\alpha$ и сравнивают ее по норме с известной погрешностью правой части $\Delta\mathbf{b}$ и с влиянием погрешности коэффициентов матрицы $\Delta\mathbf{A}\cdot\mathbf{x}$. Если α слишком велико (мало), то невязка будет намного больше (меньше) этих погрешностей. Проведя серию численных экспериментов, в качестве оптимального α выбирают то его значение, при котором $\|\mathbf{r}_\alpha\| \approx \|\Delta\mathbf{b}\| + \|\Delta\mathbf{A}\cdot\mathbf{x}\|$.

Выбор вектора $\mathbf{x}_0$ зависит от решаемой задачи. Если практических соображений для его выбора нет, то полагают $\mathbf{x}_0 = \mathbf{0}$.

§ 5.21. Задачи

1.1. Покажите, что при перестановке строк и столбцов в квадратной матрице A ее обусловленность не меняется.

1.2. Доказать, что если $\mathbf{A}$ – симметрическая положительно определенная матрица, то $(\mathbf{A}\mathbf{x},\mathbf{x})^{1/2}$ можно принять за норму вектора $\mathbf{x}$.

1.32. Показать, что модуль любого собственного значения матрицы не больше любой ее нормы.

1.4. Доказать справедливость неравенств: $\|\mathbf{x}\|_\infty \le \|\mathbf{x}\|_2 \le \|\mathbf{x}\|_1$.

1.5. Пусть $\mathbf{Q}$ – ортогональная матрица, т. е. $\mathbf{Q}^{-1} = \mathbf{Q}^T$. Показать, что тогда $\mathrm{cond}_2(\mathbf{Q}) = 1$.

1.6. Показать, что если квадратная матрица $\mathbf{A} = (a_{ij})$ порядка n имеет диагональное преобладание, то она невырождена. Данное утверждение принято называть *критерием регулярности Адамара.*

1.7. Пусть $\mathbf{A} = \begin{pmatrix} 100 & 99 \\ 99 & 98 \end{pmatrix}$. Доказать, что данная матрица имеет наибольшее число обусловленности $\mathrm{cond}_2(\mathbf{A})$ из всех невырожденных матриц второго порядка, элементами которых являются целые числа, меньшие или равные 100.

1.8. Пусть $\mathbf{A}$ – симметрическая матрица с собственными значениями $0 < \lambda_1 < < \lambda_2 < \ldots < \lambda_n$. Доказать, что $\mathrm{cond}_2(\mathbf{A}+\tau\mathbf{E})$ монотонно убывает по τ при $\tau > 0$.

1.9. Студент рассматривает возможность покупки нового компьютера и вы-

бирает между PentiumIV или AMD64. Оба компьютера тестируются на решении линейной системы:

$$\begin{cases} 100x + 99y = 199; \\ 99x + 98y = 197. \end{cases}$$

Компьютеры дают решения $\mathbf{x}_P = (1{,}99;\ 0{,}00)^T$ и $\mathbf{x}_A = (1{,}10;\ 0{,}95)^T$. Проверка точности решений производится подстановкой:

$$\mathbf{r}_P = \mathbf{b} - \mathbf{A}\mathbf{x}_P = (0{,}00;\ -0{,}01)^T,$$
$$\mathbf{r}_A = \mathbf{b} - \mathbf{A}\mathbf{x}_A = (-5{,}05;\ -5{,}00)^T.$$

Какой компьютер дает лучший результат? Почему?

1.10. Пользуясь разложениями (1.12) и (1.15), вычислить детерминант 4×4 матрицы $\mathbf{A}$ в (1.11). Сравнить полученные результаты.

1.11. Как привести матрицы а) – в) к треугольной форме, сохраняя их разреженность:

$$\text{а)} \begin{pmatrix} * & * & * & * & * \\ * & * & & & \\ * & & * & & \\ * & & & * & \\ * & & & & * \end{pmatrix}, \quad \text{б)} \begin{pmatrix} * & * & * & * & * \\ * & * & & & * \\ * & & * & & * \\ * & & & * & * \\ * & * & * & * & * \end{pmatrix}, \quad \text{в)} \begin{pmatrix} * & * & & & * \\ * & * & * & & \\ & * & * & * & \\ & & * & * & * \\ * & & & * & * \end{pmatrix}?$$

1.12. Рассмотреть матрицу $\mathbf{H} = \begin{pmatrix} -c & s \\ s & c \end{pmatrix}$, $c = \cos\varphi$, $s = \sin\varphi$:

а) показать, что матрица $\mathbf{H}$ является матрицей отражения;

б) используя матрицу $\mathbf{H}$, решить систему (1.23).

1.13. Рассмотреть матрицы

$$\mathbf{H} = \begin{pmatrix} -c & s \\ s & c \end{pmatrix}, \quad \mathbf{J} = \begin{pmatrix} c & s \\ -s & c \end{pmatrix},$$

где $c = \cos\varphi$ и $s = \sin\varphi$ для некоторого угла φ. Описать геометрический смысл умножения слева на матрицы $\mathbf{H}$ и $\mathbf{J}$. В какую сторону производится вращение (по часовой стрелке или против нее)?

1.14. Доказать, что для систем линейных уравнений второго порядка ($n = 2$) методы Якоби и Зейделя сходятся и расходятся одновременно.

1.15. Показать, что существует система уравнений третьего порядка, для которой метод Зейделя (Якоби) сходится, а метод Якоби (Зейделя) расходится.

1.16. Пусть известны все собственные значения λ невырожденной симметрической матрицы $\mathbf{A}$ порядка n. Построить итерационный метод с переменным параметром τ_k, который не более чем за n шагов приводил бы к точному решению системы $\mathbf{A}\mathbf{x} = \mathbf{b}$.

1.17. Показать, что выполнение неравенства $0 < \omega < 2$ является необходимым для сходимости метода релаксации.

1.18. Найти скорость сходимости методов Якоби, Зейделя и верхней релаксации при решении линейной системы $\mathbf{Ax} = \mathbf{b}$, где $\mathbf{A} = \begin{pmatrix} 4 & 1 \\ 1 & 4 \end{pmatrix}$. Каково оптимальное значение параметра релаксации ω? Проверить справедливость формулы Янга $\omega = 2/(1+\sqrt{1-\lambda_{max}^2})$, где λ_{max} – максимальное собственное число матрицы перехода в методе Якоби.

1.19. При каких значениях итерационного параметра τ метод простой итерации $\mathbf{x}^{(k+1)} = \mathbf{x}^{(k)} - \tau(\mathbf{Ax}^{(k)} - \mathbf{b})$ для системы уравнений $\mathbf{Ax} = \mathbf{b}$ с матрицей

$$1)\ \mathbf{A} = \begin{pmatrix} 5 & 0{,}8 & 4 \\ 2{,}5 & 3 & 0 \\ 2 & 0{,}8 & 4 \end{pmatrix}, \qquad 2)\ \mathbf{A} = \begin{pmatrix} 2 & 1 & 0{,}5 \\ 3 & 5 & 1 \\ 1 & 1 & 3 \end{pmatrix},$$

$$3)\ \mathbf{A} = \begin{pmatrix} 1 & 0{,}5 & 0{,}3 \\ 1 & 3 & 0 \\ 0{,}9 & 1 & 2 \end{pmatrix}, \quad 4)\ \mathbf{A} = \begin{pmatrix} 3 & 1{,}2 & 0{,}8 \\ 1{,}4 & 2 & 0{,}1 \\ 0{,}5 & 0{,}4 & 1 \end{pmatrix}$$

сходится при любом начальном приближении?

1.20. Пусть

$$\mathbf{A}_1 = \begin{pmatrix} 1 & -3/4 \\ -1/12 & 1 \end{pmatrix}, \quad \mathbf{A}_2 = \begin{pmatrix} 1 & -1/2 \\ -1/2 & 1 \end{pmatrix}.$$

Показать, что для матрицы $\mathbf{A}_1$ метод Якоби сходится быстрее, чем для $\mathbf{A}_2$, т. е. усиление диагонального преобладания не влечет более быструю сходимость метода Якоби.

1.21. Пусть симметрическая матрица $\mathbf{A}$ положительно определена. При каких $\alpha \in [0;1]$ метод

$$\frac{\mathbf{x}^{(k+1)} - \mathbf{x}^{(k)}}{\tau} + \mathbf{A}\big(\alpha\mathbf{x}^{(k+1)} + (1-\alpha)\mathbf{x}^{(k)}\big) = \mathbf{b}$$

сходится при любом $\tau > 0$?

Глава 6
Решение задач на собственные значения

В гл. 5 мы уже видели, что знание собственных значений матрицы очень полезно при оценке ее обусловленности и оптимизации итерационных методов решения систем линейных алгебраических уравнений. Задача отыскания собственных значений матрицы, внешне выглядящая простой, в действительности является нетривиальной. В этой главе мы рассмотрим основные итерационные методы нахождения собственных значений и соответствующих собственных векторов вещественной квадратной матрицы. Эти методы являются надежными и весьма эффективными при реальных компьютерных вычислениях. Основное внимание будет сосредоточено на симметрических матрицах, которые имеют вещественные собственные значения и ортогональную систему собственных векторов, образующих базис в $\mathbb{R}^n$.

§ 6.1. Задачи на собственные значения

Пусть имеется вещественная квадратная матрица $\mathbf{A} = \{a_{ij}\}$, $i, j = 1, 2, \ldots, n$. Числа λ, при которых уравнение

$$\mathbf{A}\mathbf{x} = \lambda\mathbf{x}, \quad \mathbf{x} \neq \mathbf{0}. \tag{6.1}$$

имеет ненулевое решение, называются *собственными значениями* матрицы $\mathbf{A}$, а ненулевые решения уравнения (6.1), соответствующие этим собственным значениям, называются *собственными векторами*.

Задачи на собственные значения разделяются на полные и частичные. В полной задаче на собственные значения требуется по матрице $\mathbf{A}$ найти все ее собственные значения и отвечающие им собственные векторы.

Различают следующие частичные задачи:

1. вычисление максимального по модулю собственного значения и соответствующего собственного вектора;

2. если собственные значения упорядочить $|\lambda_1| \geq |\lambda_2| \geq \ldots \geq |\lambda_n|$, то ставится задача нахождения $k \geq 1$ $(k < n)$ собственных значений и собственных векторов;

3. пусть задано число $a \in \mathbb{R}$; требуется вычислить собственное значение матрицы $\mathbf{A}$, ближайшее к a.

Если матрица $\mathbf{A}$ симметрична и положительно определена, то третья задача сводится к первой. В этом случае $\lambda_1 \geq \lambda_2 \geq \ldots \geq \lambda_n > 0$ и минимальное собственное значение матрицы $\mathbf{B}_a = (\mathbf{A} - a\mathbf{E})^2$ соответствует ближайшему к a. Теперь максимальное по модулю собственное значение матрицы $\|\mathbf{B}_a\|_2\mathbf{E} - \mathbf{B}_a$ будет искомым.

Всякий численный метод нахождения собственных значений матрицы нуждается в информации об их расположении. Для этого можно воспользоваться оценкой $|\lambda| \leq |\mathbf{A}\|$, применить теорему Гершгорина из гл. 1 и др.

Из уравнения (6.1) следует, что нахождение собственных чисел λ сводится к решению *векового* (или *характеристического*) уравнения

$$\det|\mathbf{A} - \lambda\mathbf{E}| = 0. \tag{6.2}$$

Раскрывая определитель, получаем эквивалентную задачу об отыскании нулей *характеристического* многочлена

$$a_0\lambda^n + a_1\lambda^{n-1} + \ldots + a_n = 0. \tag{6.3}$$

Так как алгебраические уравнения порядка $n > 4$ не имеют решения в радикалах, то отсюда сразу следует, что при больших n не существует прямых методов решения задачи (6.1). Более того нахождение собственных значений матрицы $\mathbf{A}$, основанное на решении характеристического уравнения (6.3), требует выполнения двух этапов:

1. вычисление коэффициентов $a_0, a_1, \ldots, a_n$ характеристического многочлена (6.3);

2. вычисление корней характеристического многочлена (6.3).

Покажем, что корни характеристического уравнения очень чувствительны к ошибкам округления на любом из этих этапов.

Пример 6.1. Рассмотрим матрицу Уилкинсона [28]

$$\mathbf{A}_\varepsilon = \begin{pmatrix} 20 & 20 & 0 & \ldots & 0 \\ 0 & 19 & 20 & \ldots & 0 \\ \vdots & \vdots & \ddots & \ddots & \vdots \\ 0 & 0 & \ldots & 2 & 20 \\ \varepsilon & 0 & \ldots & 0 & 1 \end{pmatrix}.$$

При $\varepsilon = 0$ собственные значения матрицы $\mathbf{A}_0$ являются целыми числами $\lambda_i = i$, $i = 1, 2, \ldots, 20$. Уравнение (6.3) для матрицы $\mathbf{A}_\varepsilon$ имеет вид:

$$\lambda_\varepsilon^{20} + a_{19}\lambda_\varepsilon^{19} + \ldots + a_1\lambda_\varepsilon + a_0 - 20^{19}\varepsilon = 0.$$

Так как $a_0 = 20!$, то выбирая теперь ε из равенства $a_0 = 20^{19}\varepsilon$, т. е. полагая $\varepsilon = 20^{-19} \cdot 20! \approx 5 \cdot 10^{-7}$, получаем вырожденную матрицу, у которой одно из ее собственных значений становится равным нулю.

Таким образом, коэффициенты характеристического многочлена очень чувствительны к ошибкам в коэффициентах исходной матрицы $\mathbf{A}$.

Пример 6.2. (Уилкинсон [28]). Рассмотрим многочлен вида

$$p(x) = (x-1)(x-2)\dots(x-20) = x^{20} - 210x^{19} + \dots .$$

Предположим, что при вычислениях мы ошиблись в коэффициенте при x^{19}, получив вместо -210 число $-210 + 2^{-23}$. Тщательное вычисление корней нового уравнения

$$p(x) + 2^{-23}x^{19} = 0$$

дает следующие результаты:

1,00000 0000	6,00000 6994	10,09526 6145 $\pm$ 0,64350 0904i
2,00000 0000	6,99969 7234	11,79363 3881 $\pm$ 1,65232 9728i
3,00000 0000	8,00726 7603	13,99235 8137 $\pm$ 2,51883 0070i
4,00000 0000	8,91725 0249	16,73073 7466 $\pm$ 2,81262 4894i
4,99999 9928	20,84690 8101	19,50243 9400 $\pm$ 1,94033 0347i

Причина такого изменения корней не в точности вычислений, а в чувствительности самого многочлена. Проанализируем, что же произошло. Так как

$$p(x, \varepsilon) = x^{20} + (-210 + \varepsilon)x^{19} + \dots ,$$

то, дифференцируя по ε, имеем

$$\frac{\partial p}{\partial \varepsilon} = \frac{\partial p}{\partial x}\frac{\partial x}{\partial \varepsilon} + \frac{\partial p}{\partial \varepsilon} = 0.$$

Тогда коэффициенты чувствительности корней

$$\frac{\partial x}{\partial \varepsilon} = -\frac{\partial p}{\partial \varepsilon}\Big/\frac{\partial p}{\partial x} = x^{19}\Big[\sum_{i=1}^{20}\prod_{\substack{j=1\\ j\neq i}}^{n}(x-j)\Big]^{-1},$$

или для i-го корня

$$\left.\frac{\partial x}{\partial \varepsilon}\right|_{x=i} = i^{19}\Big[\sum_{i=1}^{20}\prod_{\substack{j=1\\ j\neq i}}^{n}(i-j)\Big]^{-1}, \quad i = 1, 2, \dots, 20.$$

Приведенные в табл. 6.1 числа дают меру чувствительности корней к изменению коэффициента при x^{19} на ε. Наиболее чувствительными оказываются корни, наиболее сильно отклонившиеся от вещественной оси.

Таблица 6.1

Корень	$\partial x/\partial\varepsilon\|_{x=i}$	Корень	$\partial x/\partial\varepsilon\|_{x=i}$
1	$-8{,}2\times10^{-18}$	11	$-4{,}6\times10^{7}$
2	$8{,}2\times10^{-11}$	12	$2{,}0\times10^{8}$
3	$-1{,}6\times10^{-6}$	13	$-6{,}1\times10^{8}$
4	$2{,}2\times10^{-3}$	14	$1{,}3\times10^{9}$
5	$-6{,}1\times10^{-1}$	15	$-2{,}1\times10^{9}$
6	$5{,}8\times10^{1}$	16	$2{,}4\times10^{9}$
7	$-2{,}5\times10^{3}$	17	$-1{,}9\times10^{9}$
8	$6{,}0\times10^{4}$	18	$1{,}0\times10^{9}$
9	$-8{,}3\times10^{5}$	19	$-3{,}1\times10^{8}$
10	$7{,}6\times10^{6}$	20	$4{,}3\times10^{7}$

Основной вывод из приведенных примеров состоит в том, что при больших n для вычисления собственных значений матрицы $\mathbf{A}$ не следует использовать характеристический многочлен (6.3). Из соображений устойчивости разумнее искать собственные значения матрицы $\mathbf{A}$ непосредственно, применяя описываемые далее итерационные методы решения уравнения (6.1). Поэтому далее мы не будем рассматривать методы решения задачи (6.1), основанные на использовании уравнения (6.3).

Современная точка зрения [9, 27] на решение задач на собственные значения состоит в изучении так называемых *спектральных портретов*:

$$S(\varepsilon)=\{z\in\mathbb{C}:f(\lambda)\equiv\sigma_{min}(\mathbf{A}-z\mathbf{E})\le\varepsilon\},$$

где $\mathbb{C}$ – множество комплексных чисел а $\sigma_{min}(\mathbf{A}-z\mathbf{E})$ – минимальное собственное значение матрицы $\mathbf{A}-z\mathbf{E}$.

Можно показать, что собственные значения матрицы $\mathbf{A}$ содержатся в $S(\varepsilon)$. Возмущения порядка ε позволяют собственным значениям изменяться в пределах множества $S(\varepsilon)$. Поэтому ответ к задаче о вычислении собственных значений полезно давать в виде линий уровня функции $f(\lambda)$, т. е. кривых, определенных условием $f(\lambda)=\varepsilon$ при различных $\varepsilon>0$.

Тем не менее, необходимо изучить прежде всего классические методы, дающие в качестве ответа отдельные собственные значения. Во многих случаях задачи на собственные значения решаются с их помощью весьма успешно.

§ 6.2. Устойчивость задачи на собственные значения

Пусть каким-то методом найдены собственные значения $\lambda_1,\lambda_2,\ldots,\lambda_n$. Если λ_i – простой корень уравнения (6.2), то, подставляя его в равенство (6.1), соответствующий собственный вектор $\mathbf{x}_i=(x_{i1},x_{i2},\ldots,x_{in})$ можно найти как решение

однородной системы линейных алгебраических уравнений (6.1). Поскольку такое решение определяется лишь с точностью до множителя, то произвольность длины полученного вектора можно устранить путем нормировки $\mathbf{e}_i = \mathbf{x}_i / \|\mathbf{x}_i\|$.

Если λ_i и λ_j – простые корни уравнения (6.2), то соответствующие им собственные векторы $\mathbf{x}_i$ и $\mathbf{x}_j$ будут линейно независимы. Поэтому, если все корни уравнения (6.2) простые, то система собственных векторов $\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_n$ линейно независима и образует базис в $\mathbb{R}^n$. Корню λ_i кратности p могут соответствовать от одного до p линейно независимых собственных векторов. Собственные векторы, соответствующие различным кратным собственным значениям λ_i и λ_j, как и в случае простых корней, также линейно независимы. Однако полная система собственных векторов матрицы $\mathbf{A}$ при наличии кратных собственных значений, вообще говоря, может уже не образовывать базис в $\mathbb{R}^n$.

Важную роль при решении задачи (6.1) играет транспонированная матрица $\mathbf{A}^T$. Так как $\det(\mathbf{A}) = det(\mathbf{A}^T)$, то собственные значения матриц $\mathbf{A}$ и $\mathbf{A}^T$ совпадают. Их собственные векторы однако различны и связаны свойством биортогональности. Пусть уравнения $\mathbf{A}\mathbf{x}_i = \lambda_i \mathbf{x}_i$ и $\mathbf{A}^T \mathbf{y}_j = \lambda_j \mathbf{y}_j$ имеют нетривиальные решения. Поскольку первое из этих уравнений можно переписать в виде $\mathbf{x}_i^T \mathbf{A}^T = \lambda_i \mathbf{x}_i^T$, то, умножая это равенство справа на $\mathbf{y}_j$, получаем $\mathbf{x}_i^T \mathbf{A}^T \mathbf{y}_j = \lambda_i \mathbf{x}_i^T \mathbf{y}_j$. Отсюда следует, что
$\mathbf{x}_i^T \lambda_j \mathbf{y}_j = \lambda_i \mathbf{x}_i^T \mathbf{y}_j$, т. е. $(\lambda_j - \lambda_i)\mathbf{x}_i^T \mathbf{y}_j = 0$. Таким образом, собственные векторы матриц $\mathbf{A}$ и $\mathbf{A}^T$, отвечающие различным собственным значениям, взаимно ортогональны:

$$\mathbf{x}_i^T \mathbf{y}_j = 0 \quad \text{при} \quad \lambda_i \neq \lambda_j.$$

Если теперь $\mathbf{A}$ – симметрическая матрица, то $\mathbf{A} = \mathbf{A}^T$ и собственные значения матрицы $\mathbf{A}$ вещественны, а ее собственные векторы, отвечающие как различным так и кратным собственным значениям, ортогональны (поскольку $\mathbf{x}_i = \mathbf{y}_i$).

Рассмотрим вопрос об устойчивости задачи (6.1). Для простоты ограничимся случаем, когда собственные векторы матрицы образуют базис, а данное собственное значение – простое.

Пусть в матрицу $\mathbf{A}$ системы (6.1) внесено возмущение $\Delta\mathbf{A}$, которое вызвало изменение собственного значения и собственного вектора на величины $\Delta\lambda$ и $\Delta\mathbf{x}$ соответственно. Таким образом, при условии, что $\mathbf{A} + \Delta\mathbf{A}$ – невырожденная матрица, имеем возмущенную систему

$$(\mathbf{A} + \Delta\mathbf{A})(\mathbf{x} + \Delta\mathbf{x}) = (\lambda + \Delta\lambda)(\mathbf{x} + \Delta\mathbf{x})$$

или, раскрывая скобки и опуская слагаемые второго порядка малости,

$$\mathbf{A}\Delta\mathbf{x} + \Delta\mathbf{A}\mathbf{x} = \Delta\lambda\mathbf{x} + \lambda\Delta\mathbf{x}.$$

Рассмотрим два случая: $\Delta\mathbf{x} = 0$, $\Delta\lambda \neq 0$ и $\Delta\mathbf{x} \neq 0$, $\Delta\lambda = 0$. В первом случае

$$\Delta\mathbf{A}\mathbf{x}_i = \Delta\lambda_i\mathbf{x}_i \quad \text{или} \quad \mathbf{y}_i^T\Delta\mathbf{A}\mathbf{x}_i = \Delta\lambda_i\mathbf{y}_i^T\mathbf{x}_i.$$

Отсюда

$$\|\Delta\lambda_i\| \leq \omega_i\|\Delta\mathbf{A}\|, \quad \omega_i = \frac{\|\mathbf{y}_i\|\,\|\mathbf{x}_i\|}{\mathbf{y}_i^T\mathbf{x}_i} \geq 1.$$

Величину ω_i называют i-м *коэффициентом перекоса* матрицы $\mathbf{A}$. По определению $\omega_i = 1/\cos\varphi_i$, где φ_i – угол между собственным вектором $\mathbf{x}_i$ задачи (6.1) и собственным вектором $\mathbf{y}_i$ сопряженной задачи $\mathbf{A}^T\mathbf{y}_i = \lambda_i\mathbf{y}_i$. Если погрешность определения матричных коэффициентов мала и мал i-й коэффициент перекоса, то мала погрешность определения i-го собственного значения. Отметим, что в случае симметрической матрицы все коэффициенты перекоса равны единице. Поэтому задача нахождения собственных значений в этом случае является устойчивой.

Рассмотрим теперь второй случай, когда

$$\mathbf{A}\Delta\mathbf{x}_i + \Delta\mathbf{A}\mathbf{x}_i = \lambda_i\Delta\mathbf{x}_i. \tag{6.4}$$

Разложим вектор $\Delta\mathbf{x}_i$ по системе линейно независимых собственных векторов $\mathbf{x}_i$ задачи (6.1)

$$\Delta\mathbf{x}_i = \sum_{j=1}^{n}\alpha_{ij}\mathbf{x}_i. \tag{6.5}$$

Из (6.4) и (6.5) получаем

$$\Delta\mathbf{A}\mathbf{x}_i = \sum_{j\neq i}(\lambda_i - \lambda_j)\alpha_{ij}\mathbf{x}_j$$

и, следовательно,

$$|\alpha_{ij}| = \frac{|\mathbf{y}_j^T\Delta\mathbf{A}\mathbf{x}_i|}{|(\lambda_i - \lambda_j)\mathbf{y}_j^T\mathbf{x}_j|} \leq \frac{\|\mathbf{x}_i\|}{\|\mathbf{x}_j\|}\frac{\omega_j}{|\lambda_i - \lambda_j|}\|\Delta\mathbf{A}\|.$$

Поэтому, если мала погрешность определения матричных коэффициентов и малы все коэффициенты перекоса, то мала погрешность определения i-го собственного вектора, если соответствующее ему i-е собственное значение – простое. В качестве очевидного следствия из полученной оценки отметим, что если $\mathbf{A} = \mathbf{A}^T$ и все собственные значения простые, то задача отыскания собственных векторов устойчива.

Пример 6.3. Несимметрическая матрица

$$\mathbf{A} = \begin{pmatrix} p & 1 & 0 & \dots & 0 \\ 0 & p & 1 & \dots & 0 \\ \vdots & \ddots & \ddots & \ddots & \vdots \\ 0 & \dots & 0 & p & 1 \\ 0 & \dots & 0 & 0 & p \end{pmatrix},$$

где p – вещественное число, имеет только одно собственное значение $\lambda = p$ кратности n и единственный собственный вектор $\mathbf{e}_1 = (1; 0; \ldots; 0)^T$. Матрица $\mathbf{A}^T$ имеет единственный собственный вектор $\mathbf{e}_n$. Коэффициент перекоса здесь $1/(\mathbf{e}_n^T \mathbf{e}_1) = = 1/0 = \infty$.

Пример 6.4. Симметрическая матрица

$$\mathbf{A} = \begin{pmatrix} p & 1 & 0 & \ldots & 0 \\ 1 & p & 1 & \ldots & 0 \\ \vdots & \ddots & \ddots & \ddots & \vdots \\ 0 & \ldots & 1 & p & 1 \\ 0 & \ldots & 0 & 1 & p \end{pmatrix},$$

где p – вещественное число, имеет собственные значения

$$\lambda_j = p - 2\cos(j\theta), \quad \theta = \frac{\pi}{n+1}, \quad j = 1, 2, \ldots, n,$$

и соответствующие этим λ_j собственные векторы

$$\mathbf{x}_j = [\sin(j\theta); \sin(2j\theta); \ldots; \sin(nj\theta)]^T,$$

которые образуют базис в $\mathbb{R}^n$.

Выше было показано, что для симметрической матрицы задача нахождения собственных чисел и собственных векторов решается устойчиво. Произвольную вещественную матрицу $\mathbf{A}$ можно симметризовать как минимум двумя способами, рассмотрев матрицы $(\mathbf{A} + \mathbf{A}^T)/2$ или $\mathbf{A}^T\mathbf{A}$. Первый способ предпочтительнее, так как при втором способе обусловленность матрицы фактически возводится в квадрат. Это объясняет трудность решения так называемых нормальных систем линейных алгебраических уравнений вида $\mathbf{A}^T\mathbf{A}\mathbf{x} = \mathbf{A}^T\mathbf{b}$.

Наряду с требованием симметричности весьма важным является условие положительной определенности матрицы.

Теорема 6.1. *Если симметрическая матрица* $\mathbf{A}$ *с положительными диагональными элементами имеет диагональное преобладание, то она положительно определена (все ее собственные значения положительны).*

Доказательство. По теореме 5.3 (Гершгорина) имеем:

$$|a_{ii} - \lambda| \le \sum_{\substack{j=1 \\ j \ne i}}^{n} |a_{ij}| \quad \text{или} \quad -\sum_{\substack{j=1 \\ j \ne i}}^{n} |a_{ij}| \le a_{ii} - \lambda \le \sum_{\substack{j=1 \\ j \ne i}}^{n} |a_{ij}|,$$

т. е.

$$a_{ii} + \sum_{\substack{j=1 \\ j \ne i}}^{n} |a_{ij}| \ge \lambda \ge a_{ii} - \sum_{\substack{j=1 \\ j \ne i}}^{n} |a_{ij}| > 0.$$

Аналогичную оценку можно получить для любого собственного числа матрицы $\mathbf{A}$ и поэтому все они положительны. Теорема доказана.

Теорема 6.2. *Матрица* $\mathbf{A}$ *положительно определена, если и только если положительно определена матрица* $(\mathbf{A}+\mathbf{A}^T)/2$.

Доказательство. Поскольку $(\mathbf{Ax},\mathbf{x}) = (\mathbf{x},\mathbf{A}^T\mathbf{x}) = (\mathbf{A}^T\mathbf{x},\mathbf{x})$, то

$$(\mathbf{Ax},\mathbf{x}) = \frac{1}{2}\Big((\mathbf{Ax},\mathbf{x}) + (\mathbf{A}^T\mathbf{x},\mathbf{x})\Big) = \Big(\frac{\mathbf{A}+\mathbf{A}^T}{2}\mathbf{x},\mathbf{x}\Big).$$

Теорема доказана.

Матрица в примере 6.4 при $p \geq 2$ положительно определена, поскольку

$$(\mathbf{Ax},\mathbf{x}) = x_1^2 + \sum_{i=2}^{n}(x_{i-1}+x_i)^2 + x_n^2 + (p-2)\sum_{i=1}^{n} x_i^2.$$

Получим еще одну оценку влияния изменения коэффициентов матрицы на собственные значения. Говорят, что $n \times n$ матрица $\mathbf{A}$ диагонализуема, если найдется невырожденная матрица $\mathbf{V}$ такая, что

$$\mathbf{V}^{-1}\mathbf{AV} = \mathbf{\Lambda},$$

где $\mathbf{\Lambda}$ – диагональная матрица с $\lambda_1, \lambda_2, \ldots, \lambda_n$ на диагонали.

Теорема 6.3. (Бауер-Файк [8]). *Пусть* $n \times n$ *матрица* $\mathbf{A}$ *диагонализуема и* $\hat{\lambda}$ *– собственное значение возмущенной матрицы* $\mathbf{A}+\Delta\mathbf{A}$. *Тогда*

$$\min_{1\leq i\leq n} |\hat{\lambda} - \lambda_i| \leq \mathrm{cond}(\mathbf{V})\|\Delta\mathbf{A}\|.$$

Доказательство. Если $\hat{\lambda}$ – собственное значение матрицы $\mathbf{A}$, то оценка тривиальна. Пусть поэтому $\hat{\lambda} \neq \lambda_i$, $i = 1, 2, \ldots, n$ и $\mathbf{x}$ – собственный вектор матрицы $\mathbf{A}+\Delta\mathbf{A}$, отвечающий собственному значению $\hat{\lambda}$, т. е. $(\mathbf{A}+\Delta\mathbf{A})\mathbf{x} = \hat{\lambda}\mathbf{x}$. Тогда $(\hat{\lambda}\mathbf{E}-\mathbf{A})\mathbf{x} = \Delta\mathbf{Ax}$, и с учетом диагонализуемости матрицы $\mathbf{A}$ имеем $(\hat{\lambda}\mathbf{E}-\mathbf{V\Lambda V}^{-1})\mathbf{x} = \Delta\mathbf{Ax}$. Последнее равенство можно переписать в следующем виде $\mathbf{V}(\hat{\lambda}\mathbf{E}-\mathbf{\Lambda})(\mathbf{V}^{-1}\mathbf{x}) = \Delta\mathbf{Ax}$ или

$$(\hat{\lambda}\mathbf{E}-\mathbf{\Lambda})(\mathbf{V}^{-1}\mathbf{x}) = \mathbf{V}^{-1}\Delta\mathbf{AV}(\mathbf{V}^{-1}\mathbf{x}).$$

Поэтому

$$\mathbf{V}^{-1}\mathbf{x} = (\hat{\lambda}\mathbf{E}-\mathbf{\Lambda})^{-1}\mathbf{V}^{-1}\Delta\mathbf{AV}(\mathbf{V}^{-1}\mathbf{x})$$

и

$$\|\mathbf{V}^{-1}\mathbf{x}\| \leq \|(\hat{\lambda}\mathbf{E}-\mathbf{\Lambda})^{-1}\|\,\|\mathbf{V}^{-1}\|\,\|\Delta\mathbf{A}\|\,\|\mathbf{V}\|\,\|\mathbf{V}^{-1}\mathbf{x}\|.$$

Таким образом, получаем

$$1 \leq \Big[\max_{1\leq i\leq n} |(\hat{\lambda}-\lambda_i)^{-1}|\Big]\,\|\mathbf{V}^{-1}\|\,\|\mathbf{V}\|\,\|\Delta\mathbf{A}\|.$$

Отсюда следует утверждение теоремы. Теорема доказана.

Пример 6.5. Пусть

$$\mathbf{A} = \begin{pmatrix} 163 & 72 \\ -360 & -159 \end{pmatrix},\ \mathbf{V} = \begin{pmatrix} 4 & -9 \\ -9 & 20 \end{pmatrix},\ \mathbf{\Lambda} = \begin{pmatrix} 1 & 0 \\ 0 & 3 \end{pmatrix},\ \Delta\mathbf{A} = \varepsilon \begin{pmatrix} 1 & 1 \\ 0 & 0 \end{pmatrix}.$$

Тогда $\hat{\lambda} = [4 + \varepsilon \pm \sqrt{4 - 796\varepsilon + \varepsilon^2}]/2$. При $\varepsilon = 0,001$ имеем $\hat{\lambda} = 1,106; 2,895$. Изменение на 0,001 в двух элементах матрицы $\mathbf{A}$ дает изменение примерно на 0,105 в обоих собственных значениях. Здесь $\|\mathbf{V}\|_1 = \|\mathbf{V}^{-1}\|_1 = 29$, $\mathrm{cond}_1(\mathbf{V}) = 841$ и $\|\Delta\mathbf{A}\|_1 = \varepsilon$. Теорема Бауера-Файка дает завышенную оценку

$$|\hat{\lambda} - \lambda| \le \mathrm{cond}_1(\mathbf{V})\|\Delta\mathbf{A}\|_1 = 841\varepsilon = 0,841.$$

§ 6.3. Степенной метод

Рассмотрим задачу отыскания максимального собственного значения и отвечающего ему собственного вектора вещественной матрицы $\mathbf{A}$. Будем считать, что матрица $\mathbf{A}$ имеет n различных собственных значений и $|\lambda_1| > |\lambda_2| > \ldots > |\lambda_n|$ а ее собственные векторы $\mathbf{e}_i$, $i = 1, 2, \ldots, n$ образуют базис в $\mathbb{R}^n$.

Возьмем некоторый вектор начального приближения $\mathbf{x}^{(0)}$ и образуем последовательность $\mathbf{x}^{(k+1)} = \mathbf{A}\mathbf{x}^{(k)}$. Покажем, что этот подход, называемый *степенным методом*, позволяет найти максимальное по модулю собственное значение матрицы $\mathbf{A}$ и отвечающий ему собственный вектор.

Представим вектор $\mathbf{x}^{(0)}$ в виде $\mathbf{x}^{(0)} = \sum_{i=1}^n c_i\mathbf{e}_i$. Тогда

$$\mathbf{x}^{(k)} = \mathbf{A}^k\mathbf{x}^{(0)} = \sum_{i=1}^n c_i\mathbf{A}^k\mathbf{e}_i = \sum_{i=1}^n c_i\lambda_i^k\mathbf{e}_i = c_1\lambda_1^k\mathbf{e}_1 + O(|\lambda_2|^k).$$

Отсюда следуют соотношения:

$$\begin{aligned} (\mathbf{x}^{(k)}, \mathbf{x}^{(k)}) &= |c_1|^2|\lambda_1|^{2k} + O(|\lambda_1|^k|\lambda_2|^k), \\ (\mathbf{x}^{(k+1)}, \mathbf{x}^{(k)}) &= \lambda_1|c_1|^2|\lambda_1|^{2k} + O(|\lambda_1|^{k+1}|\lambda_2|^k). \end{aligned}$$

Поэтому при $c_1 \ne 0$ получаем

$$\begin{aligned} \frac{(\mathbf{x}^{(k+1)}, \mathbf{x}^{(k)})}{(\mathbf{x}^{(k)}, \mathbf{x}^{(k)})} &= \frac{\lambda_1|c_1|^2|\lambda_1|^{2k} + O(|\lambda_1|^{k+1}|\lambda_2|^k)}{|c_1|^2|\lambda_1|^{2k} + O(|\lambda_1|^k|\lambda_2|^k)} = \\ &= \lambda_1\frac{1 + O\Big(\Big|\dfrac{\lambda_2}{\lambda_1}\Big|^k\Big)}{1 + O\Big(\Big|\dfrac{\lambda_2}{\lambda_1}\Big|^k\Big)} = \lambda_1 + O\Big(\Big|\frac{\lambda_2}{\lambda_1}\Big|^k\Big). \end{aligned}$$

Кроме того, так как $\|\mathbf{x}^{(k)}\| = |c_1\lambda_1^k| + O(|\lambda_2|^k)$, то

$$\mathbf{e}_1^{(k)} = \frac{\mathbf{x}^{(k)}}{\|\mathbf{x}^{(k)}\|} = \frac{\sum_{i=1}^n c_i\lambda_i^k\mathbf{e}_i}{|c_1\lambda_1^k| + O(|\lambda_2|^k)} = \frac{c_1\lambda_1^k}{|c_1\lambda_1^k|}\mathbf{e}_1 + O\Big(\Big|\frac{\lambda_2}{\lambda_1}\Big|^k\Big),$$

т. е. в ходе итерационного процесса также получаем собственный вектор, соответствующий λ_1.

Отметим, что если $|\lambda_1| > 1$, то $\|\mathbf{x}^{(k)}\| \to \infty$ при $k \to \infty$. Поэтому при компьютерных вычислениях для достаточно большого k возможно переполнение. Если $|\lambda_1| < 1$, то $\|\mathbf{x}^{(k)}\| \to 0$ и может оказаться, начиная с некоторого k, что $\mathbf{x}^{(k)} = \mathbf{0}$. Для устранения этих явлений следует время от времени нормировать вектор $\mathbf{x}^{(k)}$, чтобы $\|\mathbf{x}^{(k)}\| = 1$.

Пример 6.6. Рассмотрим матрицу $\mathbf{A} = \begin{pmatrix} 2 & -5 & 0 \\ 2 & -9 & 0 \\ -2 & 5 & -5 \end{pmatrix}$ с собственными значениями $\lambda_1 = -8$, $\lambda_2 = -5$, $\lambda_3 = 1$, и собственными векторами

$$\mathbf{x}_1 = (-3/8; -3/4; 1)^T, \quad \mathbf{x}_2 = (0; 0; 1)^T, \quad \mathbf{x}_3 = (1; 1/5; -1/6)^T.$$

В качестве начального приближения возьмем вектор $\mathbf{x}^{(0)} = (1; 1; 1)^T$ и воспользуемся критерием остановки итераций $\|\mathbf{x}^{(k+1)} - \mathbf{x}^{(k)}\|_\infty \leq 10^{-6}$. Здесь $|\lambda_2/\lambda_1| = 0,625$ и степенной метод дает наибольшее по модулю собственное значение и отвечающий ему собственный вектор за 30 итераций.

Если начать итерации с вектора $\mathbf{x}^{(0)} = (5; 1; 0)^T$, то (при отсутствии ошибок округления) будет получено второе собственное значение $\lambda_2 = -5$ и собственный вектор $\mathbf{x}_2$. Для объяснения этого явления заметим, что $\mathbf{x}^{(0)} = c_1\mathbf{x}_1 + c_2\mathbf{x}_2 + c_3\mathbf{x}_3$, где $c_1 = 0$, $c_2 = 5/6$, $c_3 = 5$. Таким образом, $\mathbf{x}^{(0)}$ не зависит от $\mathbf{x}_1$ и степенной метод дает вектор $\mathbf{x}_2$ и отвечающее ему собственное значение.

Наконец, если $\mathbf{x}^{(0)} = (9; 0; 1)^T$, то получаем $\lambda_1 = -8$ и вектор $\mathbf{x}_1$ всего за 8 итераций. Здесь $\mathbf{x}^{(0)} = c_1\mathbf{x}_1 + c_2\mathbf{x}_2 + c_3\mathbf{x}_3$, где $c_1 = 8/3$, $c_2 = 0$, $c_3 = 10$. Вектор $\mathbf{x}^{(0)}$ не зависит от $\mathbf{x}_2$ и ошибка убывает как $|\lambda_3/\lambda_1|$ вместо $|\lambda_2/\lambda_1|$.

Для отыскания минимального собственного значения матрицы $\mathbf{A}$ и соответствующего собственного вектора применяют *обратный степенной метод*, который работает с $\mathbf{A}^{-1}$ вместо $\mathbf{A}$. Один шаг процесса имеет вид $\mathbf{x}^{(k+1)} = \mathbf{A}^{-1}\mathbf{x}^{(k)}$. Это означает, что на каждом шаге решается система $\mathbf{A}\mathbf{x}^{(k+1)} = \mathbf{x}^{(k)}$. Так как собственные значения μ_i матрицы $\mathbf{A}^{-1}$ связаны с собственными значениями λ_i матрицы $\mathbf{A}$ очевидным соотношением $\mu_i\lambda_i = 1$, то

$$|\mu_{\max}| = \frac{1}{|\lambda_{\min}|} = \frac{1}{|\lambda_n|}.$$

Поэтому в данном случае будем иметь:

$$\lim_{k\to\infty} \frac{(\mathbf{x}^{(k+1)}, \mathbf{x}^{(k)})}{(\mathbf{x}^{(k)}, \mathbf{x}^{(k)})} = \frac{1}{\lambda_n}.$$

Достаточно хорошим приближением для собственного вектора $\mathbf{e}_n$ будет вектор

$\mathbf{x}^{(k+1)}$, поскольку

$$\mathbf{x}^{(k+1)} = \sum_{i=1}^{n} \frac{c_i}{\lambda_i^{k+1}} \mathbf{e}_i = \frac{c_n}{\lambda_n^{k+1}} \Big[\sum_{i=1}^{n-1} \frac{c_i}{c_n} \Big(\frac{\lambda_n}{\lambda_i} \Big)^{k+1} \mathbf{e}_i + \mathbf{e}_n \Big] \approx \frac{c_n}{\lambda_n^{k+1}} \mathbf{e}_n.$$

В процессе итераций здесь надо также производить нормировку вектора $\mathbf{x}^{(k+1)}$ во избежание эффектов переполнения и зануления. На первой итерации следует найти LU-разложение матрицы $\mathbf{A}$, что позволит на последующих итерациях ограничиться решением систем с треугольными матрицами $\mathbf{L}\mathbf{y}^{(k)} = \mathbf{x}^{(k)}$, $\mathbf{U}\mathbf{x}^{(k+1)} = \mathbf{y}^{(k)}$. Тогда, начиная со второй итерации, для нахождения $\mathbf{x}^{(k+1)}$ требуется такое же число арифметических операций, что и в степенном методе.

Наилучшие результаты дает *обратный степенной метод со сдвигом*. Предположим, что матрица $\mathbf{A}$ заменена на $\mathbf{A} - \tau\mathbf{E}$. Тогда все собственные значения λ_i сдвигаются на величину τ и множитель, характеризующий сходимость обратного метода к λ_1, становится равным $\theta = |\lambda_1 - \tau|/|\lambda_2 - \tau|$. Поэтому, если τ выбрано так, что оно является хорошим приближением для λ_1, то θ будет очень малым и сходимость весьма ускоряется. Каждый шаг метода состоит в решении системы $(\mathbf{A} - \tau\mathbf{E})\mathbf{x}^{(k+1)} = \mathbf{x}^{(k)}$, и этому уравнению удовлетворяет вектор

$$\mathbf{x}^{(k)} = \frac{c_1\mathbf{x}_1}{(\lambda_1 - \tau)^k} + \frac{c_2\mathbf{x}_2}{(\lambda_2 - \tau)^k} + \ldots + \frac{c_n\mathbf{x}_n}{(\lambda_n - \tau)^k}.$$

Если τ близко к λ_1, то знаменатель первого члена этого представления будет близок к нулю и, следовательно, потребуется только несколько шагов, чтобы сделать первый член полностью доминирующим. В частности, если λ_1 уже каким-либо образом приближенно вычислено, то в качестве τ можно взять это вычисленное значение.

Пример 6.7. Для матрицы $\mathbf{A} = \begin{pmatrix} 2 & -5 & 0 \\ 2 & -9 & 0 \\ -2 & 5 & -5 \end{pmatrix}$ из примера 6.6 при $\mathbf{x}^{(0)} = (1;1;1)^T$ обратный степенной метод со сдвигом $\tau = -7,8$ сходится за 7 итераций. Здесь $\lambda_1 - \tau = -0,2$; $\lambda_2 - \tau = 2,8$; $\lambda_3 - \tau = 8,8$. Скорость сходимости пропорциональна $|\lambda_1 - \tau|/|\lambda_2 - \tau| \approx 0,0714$. При $\tau = -4$ обратный степенной метод сходится за 15 итераций к $\lambda_2 = -5$. Причина в том, что $\tau = -4$ ближе к $\lambda_2 = -5$ чем к $\lambda_1 = -8$.

Рассмотрим подробнее вопрос о выборе сдвигов в обратном степенном методе. Важную роль здесь играет отношение Релея $R(\mathbf{x}) = (\mathbf{A}\mathbf{x}, \mathbf{x})/(\mathbf{x}, \mathbf{x})$. Будем считать, что матрица $\mathbf{A}$ симметрична, положительно определена и $\lambda_1 > \lambda_2 > \ldots > \lambda_n$. Тогда при $\mathbf{x} = \sum_{i=1}^{n} c_i\mathbf{e}_i$ получаем

$$R(\mathbf{x}) = \frac{(\mathbf{A}\mathbf{x}, \mathbf{x})}{(\mathbf{x}, \mathbf{x})} = \frac{\lambda_1 c_1^2 + \lambda_2 c_2^2 + \ldots + \lambda_n c_n^2}{c_1^2 + c_2^2 + \ldots + c_n^2}.$$

Отсюда следует, что $\lambda_1 \geq R(\mathbf{x}) \geq \lambda_n$, причем равенство здесь достигается на векторах $\mathbf{e}_1$ и $\mathbf{e}_n$. Поэтому

$$\lambda_1 = \max_{\mathbf{x}\neq\mathbf{0}} \frac{(\mathbf{A}\mathbf{x},\mathbf{x})}{(\mathbf{x},\mathbf{x})}, \quad \lambda_n = \min_{\mathbf{x}\neq\mathbf{0}} \frac{(\mathbf{A}\mathbf{x},\mathbf{x})}{(\mathbf{x},\mathbf{x})}.$$

Таким образом, в качестве сдвигов τ в обратном степенном методе можно использовать величины $R(\mathbf{x}^{(k)})$.

§ 6.4. Метод исчерпывания

Пусть $\mathbf{A}$ – симметрическая матрица, которая имеет n различных собственных значений таких, что $|\lambda_1| > |\lambda_2| > \ldots > |\lambda_n|$. Ее собственные векторы $\mathbf{e}_i$, $i = 1, 2, \ldots, n$ образуют ортонормированный базис в $\mathbb{R}^n$.

Предположим, что максимальное собственное значение λ_1 и отвечающий ему собственный вектор $\mathbf{e}_1$ уже найдены каким-либо итерационным методом (например, степенным методом). В силу ортонормированности собственных векторов матрицы $\mathbf{A}$ можно записать:

$$\mathbf{e}_i^T\mathbf{e}_j = 0 \quad \text{при} \quad i \neq j, \qquad \mathbf{e}_i^T\mathbf{e}_i = 1.$$

Если образовать новую матрицу $\mathbf{A}^* = \mathbf{A} - \lambda_1\mathbf{e}_1\mathbf{e}_1^T$, то в силу выписанных равенств ее собственные значения и собственные векторы будут связаны соотношениями:

$$\begin{aligned} \mathbf{A}^*\mathbf{e}_1 &= \mathbf{A}\mathbf{e}_1 - \lambda_1\mathbf{e}_1\mathbf{e}_1^T\mathbf{e}_1 = \mathbf{A}\mathbf{e}_1 - \lambda_1\mathbf{e}_1 = 0; \\ \mathbf{A}^*\mathbf{e}_i &= \mathbf{A}\mathbf{e}_i - \lambda_1\mathbf{e}_1\mathbf{e}_1^T\mathbf{e}_i = \lambda_i\mathbf{e}_i; \quad i = 2, 3, \ldots, n. \end{aligned}$$

Следовательно, матрица $\mathbf{A}^*$ имеет собственные значения $0, \lambda_2, \ldots, \lambda_n$ и те же самые собственные векторы $\mathbf{e}_i$, $i = 1, 2, \ldots, n$. Так как теперь максимальным собственным значением является λ_2, то к матрице $\mathbf{A}^*$ можно опять применить итерационный метод и найти λ_2 и $\mathbf{e}_2$. После этого образуется матрица $\mathbf{A}^{**} = \mathbf{A}^* - \lambda_2\mathbf{e}_2\mathbf{e}_2^T$, которая будет иметь собственные значения $0, 0, \lambda_3, \ldots, \lambda_n$ и те же самые собственные векторы, что и матрица $\mathbf{A}$, и т. д.

Теоретически на этом пути можно найти все собственные значения и собственные векторы матрицы $\mathbf{A}$. Однако при реальных компьютерных вычислениях выполнение каждого шага приводит к возникновению погрешностей при вычислении собственных векторов, которые будут сказываться на точности определения следующего собственного вектора и вызывать накопление ошибок. Поэтому описанный метод вряд ли применим для нахождения более чем трех собственных значений, начиная с наибольшего или наименьшего.

Степенной метод, обратный степенной метод со сдвигами и метод исчерпывания позволяют решить частичную проблему собственных значений, когда требуется найти только некоторые собственные значения и соответствующие им собственные векторы матрицы $\mathbf{A}$. Перейдем теперь к рассмотрению методов решения

полной проблемы собственных значений. Так как легче всего находятся собственные значения диагональной, трехдиагональной, треугольной и почти треугольной матриц, то одна из основных идей этих методов состоит в преобразовании матрицы к одной из этих простых форм при помощи преобразований подобия, которые не меняют собственных значений матрицы и по определенному закону преобразуют ее собственные векторы.

§ 6.5. Метод вращений Якоби

Этим методом решается полная проблема собственных значений вещественной симметрической матрицы. Метод Якоби позволяет привести матрицу к диагональному виду путем зануления наибольших по модулю элементов, стоящих вне главной диагонали. К сожалению, приведение к строго диагональному виду требует бесконечно большого числа шагов, так как образование нового нулевого элемента на месте одного из элементов матрицы почти всегда ведет к появлению ненулевого элемента там, где ранее был нуль. Этого следует ожидать, так как в общем случае мы не можем найти корни многочлена за конечное число шагов. На практике метод Якоби рассматривают как итерационную процедуру, которая в принципе позволяет достаточно близко подойти к диагональной форме, чтобы это преобразование можно было считать законченным.

6.5.1. Вращение плоскости. Плоское вращение на угол θ достигается умножением векторов слева на матрицу

$$\mathbf{R}(\theta) = \begin{pmatrix} c & -s \\ s & c \end{pmatrix}, \quad c = \cos\theta, \quad s = \sin\theta.$$

Соответствующее преобразование подобия симметрической матрицы $\mathbf{A}$ размерности два имеет вид:

$$\mathbf{RAR}^T = \begin{pmatrix} c & -s \\ s & c \end{pmatrix} \begin{pmatrix} \alpha & \gamma \\ \gamma & \beta \end{pmatrix} \begin{pmatrix} c & s \\ -s & c \end{pmatrix} =$$
$$= \begin{pmatrix} \alpha c^2 - 2\gamma sc + \beta s^2 & (c^2 - s^2)\gamma + (\alpha - \beta)sc \\ (c^2 - s^2)\gamma + (\alpha - \beta)sc & \alpha s^2 + 2\gamma sc + \beta c^2 \end{pmatrix}.$$

Новая матрица будет диагональной, если

$$\operatorname{tg} 2\theta = \frac{\sin 2\theta}{\cos 2\theta} = \frac{2sc}{c^2 - s^2} = \frac{2\gamma}{\beta - \alpha}.$$

Нет необходимости вычислять θ в явном виде. Пусть

$$\delta = |\beta - \alpha|/2, \quad \nu = \sqrt{\gamma^2 + \delta^2}.$$

Пользуясь стандартными тригонометрическими тождествами, находим

$$c^2 = \frac{1}{2}(1 + \delta/\nu), \quad s^2 = \frac{1}{2}(1 - \delta/\nu).$$

Выражение для s^2 представляет собой классический пример формулы, опасной при вычислениях с конечной точностью. Если θ мало, то δ/ν близко к 1, и формула была бы хороша, если бы легко было вычислить δ/ν с запасом точности. Имеется несколько устойчивых способов вычисления $\cos\theta$ и $\sin\theta$ по данной информации.

а) Способ Рутисхаузера [22]. Посредством тригонометрических тождеств $\operatorname{tg} 2\theta$ можно выразить через $t = \operatorname{tg}\theta$:

$$\frac{1 - t^2}{2t} = \operatorname{ctg} 2\theta = \frac{\beta - \alpha}{2\gamma} \equiv \zeta.$$

Таким образом, t – наименьший по абсолютной величине корень уравнения

$$t^2 + 2\zeta t - 1 = 0$$

и, следовательно,

$$t = \operatorname{sign}(\zeta)/(|\zeta| + \sqrt{1 + \zeta^2}).$$

Тогда

$$c = 1/\sqrt{1 + t^2}; \quad s = ct.$$

Навязывая величине c положительный знак, получаем вращение с углом $|\theta| \leq \pi/4$. Есть и другое решение уравнения $\operatorname{ctg} 2\theta = \zeta$. Это – угол $\pi/2 + \theta$, который нам однако не потребуется. Применение тригонометрии показывает также, что новая матрица имеет вид:

$$\begin{pmatrix} \alpha - \gamma t & 0 \\ 0 & \beta + \gamma t \end{pmatrix}.$$

б) Способ Уилкинсона-Воеводина [28]. При $\alpha = \beta$ берем $c = s = \sqrt{2}/2$ иначе полагаем $z = \max(|\gamma|, |\alpha - \beta|/2)$, $x_1 = \gamma/z$, $y_1 = (\alpha - \beta)/(2z)$ и тогда

$$c = \left(\frac{1}{2}\left(1 + \frac{|y_1|}{\sqrt{x_1^2 + y_1^2}}\right)\right)^{1/2}; \quad s = \frac{\operatorname{sign}(x_1 y_1)|x_1|}{2c\sqrt{x_1^2 + y_1^2}}.$$

6.5.2. Вращения Якоби. Пусть $\mathbf{A}$ – симметрическая матрица размерности n. Применим изложенную выше технику к этой матрице. Стратегия метода Якоби состоит в том, что на m-м шаге отыскивается максимальный по модулю внедиагональный элемент матрицы $\mathbf{A}^{(m-1)}$ ($\mathbf{A}^{(0)} = \mathbf{A}$), который для определенности будем считать лежащим ниже главной диагонали:

$$|a_{kl}^{(m-1)}| = \max_{i,j} |a_{ij}^{(m-1)}|, \quad k > l, \quad 1 \leq j \leq i - 1, \quad i = 2, 3, \ldots, n,$$

и с помощью плоских вращений $\mathbf{R}_m(k,l,\theta)$ этот элемент зануляется. Если положить $\alpha = a_{kk}^{(m-1)}$, $\beta = a_{ll}^{(m-1)}$, $\gamma = a_{kl}^{(m-1)}$, то приведенная формула для выбора ζ и формулы для c и s неявно определяют угол θ такой, что элементы $a_{kl}^{(m)}$ и $a_{lk}^{(m)}$ матрицы $\mathbf{A}^{(m)} = \mathbf{R}_m(k,l,\theta)\mathbf{A}^{(m-1)}\mathbf{R}_m(k,l,\theta)^T$ суть нули; именно

$$\operatorname{tg} 2\theta = 2a_{kl}^{(m-1)}/(a_{ll}^{(m-1)} - a_{kk}^{(m-1)}).$$

При таком выборе θ матрица $\mathbf{R}_m(k,l,\theta)$ называется *матрицей якобиева вращения*, а соответствующее подобное преобразование – *якобиевым вращением*. Оно зануляет элемент $a_{kl}^{(m)}$ и, поскольку θ фиксировано заданием ζ, мы можем писать $\mathbf{R}_m(k,l)$ вместо $\mathbf{R}_m(k,l,\theta)$, не впадая в двусмысленность.

Новым при $n > 2$ будет то, что в строках и столбцах с номерами k и l матрицы $\mathbf{A}^{(m)}$ имеются другие внедиагональные элементы, которые подвергаются преобразованию. Так при $i \neq k,l$

$$\begin{aligned} a_{ki}^{(m)} &= c \cdot a_{ki}^{(m-1)} - s \cdot a_{li}^{(m-1)} = a_{ik}^{(m)}, \\ a_{li}^{(m)} &= s \cdot a_{ki}^{(m-1)} + c \cdot a_{li}^{(m-1)} = a_{il}^{(m)}. \end{aligned}$$

Остальные элементы вычисляются по формулам:

$$\begin{aligned} a_{kk}^{(m)} &= c^2 \cdot a_{kk}^{(m-1)} - 2cs \cdot a_{kl}^{(m-1)} + s^2 \cdot a_{ll}^{(m-1)}; \\ a_{ll}^{(m)} &= s^2 \cdot a_{kk}^{(m-1)} + 2cs \cdot a_{kl}^{(m-1)} + c^2 \cdot a_{ll}^{(m-1)}; \\ a_{kl}^{(m)} &= (c^2 - s^2)a_{kl}^{(m-1)} + cs \cdot (a_{kk}^{(m-1)} - a_{ll}^{(m-1)}) = a_{lk}^{(m)}. \end{aligned}$$

Отметим еще, что при $i \neq k,l$

$$(a_{ki}^{(m)})^2 + (a_{li}^{(m)})^2 = (a_{ki}^{(m-1)})^2 + (a_{li}^{(m-1)})^2.$$

Докажем сходимость метода Якоби. Пусть $\omega(\mathbf{A}^{(m)}) = \sum_{i \neq j} a_{ij}^2$. Тогда в силу соотношений между элементами матриц $\mathbf{A}^{(m)}$ и $\mathbf{A}^{(m-1)}$ получаем $\omega(\mathbf{A}^{(m)}) = \omega(\mathbf{A}^{(m-1)}) - 2(a_{kl}^{(m-1)})^2$. Поскольку $a_{kl}^{(m-1)}$ – максимальный по модулю внедиагональный элемент матрицы $\mathbf{A}^{(m-1)}$, то

$$\omega(\mathbf{A}^{(m-1)}) \leq n(n-1)(a_{kl}^{(m-1)})^2$$

и, следовательно,

$$(a_{kl}^{(m-1)})^2 \geq \frac{1}{n(n-1)}\omega(\mathbf{A}^{(m-1)}).$$

С помощью этого неравенства получаем

$$\omega(\mathbf{A}^{(m)}) \;=\; \omega(\mathbf{A}^{(m-1)}) - 2(a_{kl}^{(m-1)})^2 \leq \omega(\mathbf{A}^{(m-1)}) - \frac{2}{n(n-1)}\omega(\mathbf{A}^{(m-1)}) =$$

$$= \quad q\omega(\mathbf{A}^{(m-1)}), \quad q = 1 - \frac{2}{n(n-1)},$$

где $0 \le q < 1$ при $n \ge 2$.

Таким образом,

$$\omega(\mathbf{A}^{(m)}) \le q\omega(\mathbf{A}^{(m-1)}) \le \ldots \le q^m\omega(\mathbf{A}^{(0)}) = q^m\omega(\mathbf{A}),$$

что означает сходимость метода Якоби со скоростью, не меньшей скорости сходимости геометрической прогрессии со знаменателем q.

Контроль точности вычислений осуществляется следующим образом. Простые расчеты показывают, что сумма квадратов диагональных элементов возрастает от итерации к итерации, т. е. если $T_m = \sum_{i=1}^{n}(a_{ii}^{(m)})^2$, то $T_m = T_{m-1} + 2(a_{kl}^{(m-1)})^2$. Средняя величина диагонального элемента может быть вычислена как $(T_m/n)^{1/2}$. Итерации оканчиваются, когда все внедиагональные элементы матрицы $\mathbf{A}^{(m)}$ становятся достаточно малы $|a_{kl}^{(m)}| = \max_{i,j}|a_{ij}^{(m)}| < \varepsilon(T_m/n)^{1/2}$, где $\varepsilon > 0$ – порог малости. Тогда элементы $a_{ii}^{(m)}$ являются приближениями к собственным значениям λ_s матрицы $\mathbf{A}$ и можно показать [3], что

$$|a_{ii}^{(m)} - \lambda_s|^2 \le \omega(\mathbf{A}^{(m)}).$$

С рабочей точностью можно также записать

$$\mathbf{R}_m \ldots \mathbf{R}_2\mathbf{R}_1\mathbf{A}\mathbf{R}_1^T\mathbf{R}_2^T \ldots \mathbf{R}_m^T = \mathbf{\Lambda},$$

где $\mathbf{\Lambda}$ – диагональная матрица собственных значений матрицы $\mathbf{A}$. Отсюда следует, что

$$\mathbf{A}\mathbf{V} = \mathbf{V}\mathbf{\Lambda}, \quad \mathbf{V} = \mathbf{R}_1^T\mathbf{R}_2^T \ldots \mathbf{R}_m^T,$$

или, что то же самое,

$$\mathbf{A}\mathbf{v}_i = \lambda_i\mathbf{v}_i, \quad i = 1, 2, \ldots, n,$$

где $\mathbf{v}_i$ – i-й столбец матрицы $\mathbf{V}$. Таким образом, столбцы матрицы $\mathbf{V}$ являются собственными векторами матрицы $\mathbf{A}$, отвечающими собственным значениям λ_i.

Пример 6.8. Применяя вращения Якоби к матрице $\begin{pmatrix} 1 & 1 & 1 & 1 \\ 1 & 2 & 3 & 4 \\ 1 & 3 & 6 & 10 \\ 1 & 4 & 10 & 20 \end{pmatrix}$, имеем

m	0	1	2	3	4
$\omega(\mathbf{A}^{(m)})$	10^2	10^1	10^{-2}	10^{-11}	10^{-17}

Рассмотренный метод подобных преобразований с ортогональной матрицей требует выбора среди внедиагональных элементов наибольшего по модулю, для

чего необходимо выполнить приблизительно n^2 операций. Можно указать много других способов выбора максимального по модулю внедиагонального элемента. Например, можно вычислить величины

$$S_i(\mathbf{A}) = \sum_{j \neq i} a_{ij}^2, \quad i = 1, 2, \ldots, n$$

и если $S_i(\mathbf{A})$ является максимальной, то в качестве искомого элемента берется наибольший по модулю элемент строки i. При таком правиле выбора максимального внедиагонального элемента необходимо будет выполнить приблизительно $2n$ операций. Оценка скорости сходимости метода Якоби при этом сохраняется.

Метод вращений является одним из самых удобных итерационных методов для определения собственных значений и собственных векторов симметрических матриц. Он прост по вычислительной схеме, быстро сходится, кратные и близкие собственные значения не вызывают никаких затруднений.

§ 6.6. Метод вращений Гивенса

Метод вращений Гивенса основан на преобразовании подобия, аналогичном применяемому в методе Якоби. Однако в отличие от метода Якоби полученные нулевые элементы остаются нулями и дальше. Поэтому метод Гивенса требует конечного числа преобразований и связан с меньшими затратами машинного времени по сравнению с методом Якоби. Однако в этом методе произвольная вещественная $n \times n$ матрица $\mathbf{A} = (a_{ij})$ приводится к матрице в верхней форме Хессенберга, которая отличается от верхней треугольной матрицы наличием одной ненулевой поддиагонали, т. е. $a_{ij} = 0$ для всех $i > j + 1$. Симметрическая матрица приводится не к диагональному, а к трехдиагональному виду.

Преобразование будем проводить за $(n-2)$ цикла. Каждый цикл включает $n-j-1$ преобразований подобия с помощью матриц вращения. Рассмотрим 1-й цикл. Он заключается в занулении элементов, стоящих в позициях $(3,1), (4,1), \ldots, (n,1)$, за счет элемента в позиции $(2,1)$. Пусть $\mathbf{A}^{(2,1)} \equiv \mathbf{A}$. Формируем последовательность подобных матриц

$$\mathbf{A}^{(k,1)} = \mathbf{R}(2,k)\mathbf{A}^{(k-1,1)}\mathbf{R}(2,k)^T, \quad 3 \leq k \leq n.$$

Элементы матрицы вращения $\mathbf{R}(2,k)$ в позициях (i,j), $i,j = 2,k$ образуются из условия зануления элемента

$$a_{k,1}^{(k,1)} = -s a_{21}^{(k-1,1)} + c a_{k1}^{(k-1,1)} = 0$$

в новой матрице $\mathbf{A}^{(k,1)}$, что с учетом равенства $s^2 + c^2 = 1$ дает

$$c = a_{21}^{(k-1,1)}/p, \quad s = a_{k1}^{(k-1,1)}/p, \quad p = ((a_{21}^{(k-1,1)})^2 + (a_{k1}^{(k-1,1)})^2)^{1/2} \neq 0$$

и $c = 1$, $s = 0$ в противном случае. Отметим, что умножение на матрицу вращения $\mathbf{R}(2,k)^T$ справа меняет 2-й и k-й столбцы, но не меняет 1-й столбец. После $n-2+n-3+\ldots+1 = (n-1)(n-2)/2$ таких преобразований матрица $\mathbf{A}$ приводится к матрице в верхней форме Хессенберга

$$\mathbf{A}^{(n,n-2)} = \mathbf{Q}\mathbf{A}\mathbf{Q}^T, \quad \mathbf{Q} = \mathbf{R}(n-1,n)\ldots\mathbf{R}(2,3).$$

Матрица $\mathbf{A}^{(n,n-2)}$ имеет те же собственные значения, что и матрица $\mathbf{A}$, а их собственные векторы связаны соотношением $\mathbf{u}_i = \mathbf{Q}\mathbf{v}_i$, $i = 1,2,\ldots,n$.

Если матрица $\mathbf{A}$ – симметрическая, то матрица в верхней форме Хессенберга также будет симметрической

$$(\mathbf{A}^{(n,n-2)})^T = (\mathbf{Q}\mathbf{A}\mathbf{Q}^T)^T = \mathbf{Q}\mathbf{A}\mathbf{Q}^T = \mathbf{A}^{(n,n-2)},$$

и поэтому трехдиагональной. Для случая $n = 5$ и симметрической матрицы $\mathbf{A}$ процесс Гивенса иллюстрируется приводимой схемой:

$$\begin{pmatrix} * & * & * & * & * \\ * & * & * & * & * \\ * & * & * & * & * \\ * & * & * & * & * \\ * & * & * & * & * \end{pmatrix} \to \begin{pmatrix} * & * & 0 & 0 & 0 \\ * & * & * & * & * \\ 0 & * & * & * & * \\ 0 & * & * & * & * \\ 0 & * & * & * & * \end{pmatrix} \to \begin{pmatrix} * & * & 0 & 0 & 0 \\ * & * & * & 0 & 0 \\ 0 & * & * & * & * \\ 0 & 0 & * & * & * \\ 0 & 0 & * & * & * \end{pmatrix} \to \begin{pmatrix} * & * & 0 & 0 & 0 \\ * & * & * & 0 & 0 \\ 0 & * & * & * & 0 \\ 0 & 0 & * & * & * \\ 0 & 0 & 0 & * & * \end{pmatrix}.$$

При условии, что мы можем найти собственные значения трехдиагональной матрицы сравнительно быстро, метод Гивенса будет работать намного быстрее метода Якоби.

§ 6.7. Метод отражений Хаусхолдера

В методе вращений Гивенса на каждом шаге получаем один нуль. Рассмотрим теперь метод, который позволяет на каждом шаге получать целый столбец (строку) нулей. Воспользуемся матрицами отражения из гл. 5, применявшимися для получения QR-разложения матрицы $\mathbf{A}$.

Пусть $\mathbf{A}$ – вещественная $n \times n$ матрица. Используя преобразования подобия, образуем последовательность матриц:

$$\mathbf{A}^{(k)} = \mathbf{H}_k^{-1}\mathbf{A}^{(k-1)}\mathbf{H}_k; \quad k = 1,2,\ldots,n-2; \quad \mathbf{A}^{(0)} = \mathbf{A}; \quad \mathbf{A}^{(k)} = (a_{ij}^{(k)}).$$

Здесь матрица подобия $\mathbf{H}_k$ образуется из единичной матрицы по правилу

$$\begin{aligned} \mathbf{H}_k &= \mathbf{E} - 2\mathbf{w}_k\mathbf{w}_k^T, \quad \mathbf{w}_k = \frac{1}{c}(0;\ldots;0;a_{k+1,k}^{(k-1)}+\sigma;a_{k+2,k}^{(k-1)};\ldots;a_{n,k}^{(k-1)})^T, \\ c^2 &= 2\sigma(a_{k+1,k}^{(k-1)}+\sigma), \quad \sigma^2 = \sum_{i=k+1}^{n}(a_{ik}^{(k-1)})^2. \end{aligned}$$

Умножение на матрицу $\mathbf{H}_k^{-1}$ позволяет занулить элементы k-го столбца матрицы $\mathbf{A}^{(k-1)}$, стоящие в позициях $(k+1,k),\ldots,(n,k)$. Последующее умножение на матрицу $\mathbf{H}_k$ справа не меняет элементы первых k столбцов. Поэтому после выполнения $(n-2)$-х шагов процесса Хаусхолдера матрица $\mathbf{A}$ приводится к матрице в верхней форме Хессенберга

$$\mathbf{A}^{(n-2)} = \mathbf{H}_{n-2}^{-1}\ldots\mathbf{H}_1^{-1}\mathbf{A}\mathbf{H}_1\ldots\mathbf{H}_{n-2} = \mathbf{Q}^{-1}\mathbf{A}\mathbf{Q}.$$

Здесь $\mathbf{Q}$ – ортогональная матрица как произведение ортогональных матриц $\mathbf{H}_k$.

В частности, если $\mathbf{A}$ – симметрическая матрица, то

$$(\mathbf{A}^{(n-2)})^T = (\mathbf{Q}^{-1}\mathbf{A}\mathbf{Q})^T = \mathbf{Q}^{-1}\mathbf{A}\mathbf{Q} = \mathbf{A}^{(n-2)}$$

и, следовательно, $\mathbf{A}^{(n-2)}$ – трехдиагональная матрица.

Пусть $\mathbf{v}_i$ – собственный вектор матрицы $\mathbf{A}^{(n-2)}$, отвечающий собственному значению λ_i, т. е.

$$\mathbf{A}^{(n-2)}\mathbf{v}_i = \mathbf{Q}^{-1}\mathbf{A}\mathbf{Q}\mathbf{v}_i = \lambda_i\mathbf{v}_i.$$

Тогда

$$\mathbf{A}(\mathbf{Q}\mathbf{v}_i) = \lambda_i(\mathbf{Q}\mathbf{v}_i)$$

и, следовательно, собственные значения подобных матриц $\mathbf{A}$ и $\mathbf{A}^{(n-2)}$ совпадают, а их собственные векторы связаны соотношением $\mathbf{u}_i = \mathbf{Q}\mathbf{v}_i$.

Пример 6.9. Пусть матрицу

$$\mathbf{A} = \begin{pmatrix} 1 & 0 & 1 \\ 0 & 1 & 1 \\ 1 & 1 & 0 \end{pmatrix}$$

требуется привести к верхней форме Хессенберга.

Здесь $\sigma^2 = a_{21}^2 + a_{31}^2 = 1$, $c^2 = 2\sigma(a_{21}^2 + \sigma) = 2$ и $\mathbf{w}_1 = \frac{1}{\sqrt{2}}(0;1;1)^T$. Поэтому

$$\mathbf{H}_1 = \mathbf{E} - 2\mathbf{w}_1\mathbf{w}_1^T = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 0 & -1 \\ 0 & -1 & 0 \end{pmatrix}.$$

Используя преобразование подобия, получаем

$$\mathbf{A}^{(1)} = \mathbf{H}_1^{-1}\mathbf{A}\mathbf{H}_1 = \begin{pmatrix} 1 & -1 & 0 \\ -1 & 0 & 1 \\ 0 & 1 & 1 \end{pmatrix}.$$

Так как исходная матрица $\mathbf{A}$ была симметрической, то полученная матрица в верхней форме Хессенберга $\mathbf{A}^{(1)}$ оказывается трехдиагональной.

§ 6.8. QR-алгоритм

Этот алгоритм позволяет найти все собственные значения вещественной матрицы $\mathbf{A}$ размерности $n \times n$ и основывается на представлении последней в виде произведения $\mathbf{A} = \mathbf{QR}$, где $\mathbf{Q}$ – ортогональная матрица, а $\mathbf{R}$ – верхняя треугольная матрица.

6.8.1. Разложение $\mathbf{A} = \mathbf{QR}$. В общем случае, применяя цепочку преобразований Хаусхолдера с матрицами отражения:

$$\begin{aligned} \mathbf{H}_k &= \mathbf{E} - 2\mathbf{w}_k\mathbf{w}_k^T, \quad \mathbf{w}_k = \frac{1}{c}(0;\ldots;0;a_{k,k}^{(k-1)}+\sigma;a_{k+1,k}^{(k-1)};\ldots;a_{n,k}^{(k-1)})^T, \\ c^2 &= 2\sigma(a_{k,k}^{(k-1)}+\sigma), \quad \sigma^2 = \sum_{i=k}^{n}(a_{ik}^{(k-1)})^2, \end{aligned}$$

матрицу $\mathbf{A}$ приводим к верхней треугольной форме:

$$\mathbf{R} = \mathbf{H}_{n-1}^{-1}\ldots\mathbf{H}_2^{-1}\mathbf{H}_1^{-1}\mathbf{A} = \mathbf{Q}^{-1}\mathbf{A}.$$

Здесь уже не допускается существование ненулевой поддиагонали, так как у нас теперь нет множителей $\mathbf{H}_k$ справа, «портящих» уже имеющиеся нули. В результате получаем:

$$\mathbf{A} = \mathbf{QR}, \quad \mathbf{Q} = \mathbf{H}_1\mathbf{H}_2\ldots\mathbf{H}_{n-1}.$$

Такое разложение требует $O(n^3)$ арифметических операций.

Если матрица $\mathbf{A}$ заранее была приведена к верхней форме Хессенберга, то, используя плоские вращения, по методу Гивенса можно исключить элементы ненулевой поддиагонали $a_{k+1,k}$, $k = 1, 2, \ldots, n-1$, и получить опять представление матрицы $\mathbf{A}$ в виде $\mathbf{A} = \mathbf{QR}$. Первое из таких преобразований при $n = 5$ имеет вид:

$$\mathbf{Q}_{21}\mathbf{A} = \begin{pmatrix} c & -s & 0 & 0 & 0 \\ s & c & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} a_{11} & * & * & * & * \\ a_{21} & * & * & * & * \\ 0 & * & * & * & * \\ 0 & 0 & * & * & * \\ 0 & 0 & 0 & * & * \end{pmatrix}.$$

Элемент в позиции $(2,1)$ этого произведения $sa_{11} + ca_{21}$ зануляется соответствующим заданием s и c. Затем $\mathbf{Q}_{32}$ выбирается таким же образом, чтобы занулить элемент матрицы $\mathbf{Q}_{32}\mathbf{Q}_{21}\mathbf{A}$ в позиции $(3,2)$ и т. д. После $n-1$ элементарных вращений получаем верхнюю треугольную матрицу

$$\mathbf{R} = \mathbf{Q}_{nn-1}\ldots\mathbf{Q}_{32}\mathbf{Q}_{21}\mathbf{A} \quad \rightarrow \quad \mathbf{A} = \mathbf{QR}.$$

Такое разложение требует $O(n^2)$ арифметических операций.

6.8.2. QR-алгоритм. Пусть $\mathbf{A}^{(0)} = \mathbf{A} = \mathbf{QR} = \mathbf{Q}_0\mathbf{R}_0$. Образуем последовательность матриц:

$$\mathbf{A}^{(k)} = \mathbf{Q}_k\mathbf{R}_k, \quad \mathbf{A}^{(k+1)} = \mathbf{R}_k\mathbf{Q}_k; \quad k = 0, 1, \ldots$$

Здесь матрица $\mathbf{A}^{(k+1)}$ получается из $\mathbf{A}^{(k)}$ перестановкой местами сомножителей $\mathbf{Q}_k$ и $\mathbf{R}_k$. Эта операция, в действительности, означает применение преобразования подобия

$$\mathbf{A}^{(k+1)} = \mathbf{R}_k\mathbf{Q}_k = \mathbf{Q}_k^{-1}\mathbf{Q}_k\mathbf{R}_k\mathbf{Q}_k = \mathbf{Q}_k^{-1}\mathbf{A}^{(k)}\mathbf{Q}_k.$$

Последовательность матриц $\mathbf{A}^{(k)}$, $k = 0, 1, \ldots$, задает QR-алгоритм решения полной проблемы собственных значений, предложенный В. Н. Кублановской и Д. Френсисом [28]. Приведем без доказательства следующий результат [8].

Теорема 6.4. *Пусть матрица* $\mathbf{A}$ *размерности* $n \times n$ *имеет собственные значения* $\lambda_1, \lambda_2, \ldots, \lambda_n$, *удовлетворяющие условию*

$$|\lambda_1| > |\lambda_2| > \ldots > |\lambda_n| > 0.$$

Тогда последовательность матриц $\{\mathbf{A}^{(k)}\}$ *в QR-алгоритме сходится к верхней треугольной матрице* $\mathbf{T}$, *диагональные элементы которой являются собственными значениями матрицы* $\mathbf{A}$. *Если* $\mathbf{A}$ *– симметрическая, то* $\mathbf{T}$ *– диагональная матрица. Имеет место оценка*

$$\|\mathbf{T} - \mathbf{A}^{(k)}\| \le c \max_i \left|\frac{\lambda_{i+1}}{\lambda_i}\right|.$$

Принципиальным моментом в QR-алгоритме является тот факт, что если матрица $\mathbf{A}$ была заранее приведена к матрице в верхней форме Хессенберга, то эта почти треугольная форма сохраняется при преобразованиях QR-алгоритма. Как следствие каждый шаг QR-алгоритма будет требовать только $O(n^2)$ арифметических операций. Проиллюстрируем это свойство на примере матрицы 5×5. Пусть

$$\mathbf{A} = \mathbf{A}_0 = \begin{pmatrix} * & * & * & * & * \\ * & * & * & * & * \\ 0 & * & * & * & * \\ 0 & 0 & * & * & * \\ 0 & 0 & 0 & * & * \end{pmatrix} = \mathbf{Q}_0\mathbf{R}_0 = \mathbf{Q}_0 \begin{pmatrix} * & * & * & * & * \\ 0 & * & * & * & * \\ 0 & 0 & * & * & * \\ 0 & 0 & 0 & * & * \\ 0 & 0 & 0 & 0 & * \end{pmatrix},$$

где точками обозначены ненулевые элементы. Отсюда следует, что матрица $\mathbf{R}_0^{-1}$ также будет верхней треугольной. Поэтому

$$\mathbf{Q}_0 = \mathbf{A}\mathbf{R}_0^{-1} = \begin{pmatrix} * & * & * & * & * \\ * & * & * & * & * \\ 0 & * & * & * & * \\ 0 & 0 & * & * & * \\ 0 & 0 & 0 & * & * \end{pmatrix} \begin{pmatrix} * & * & * & * & * \\ 0 & * & * & * & * \\ 0 & 0 & * & * & * \\ 0 & 0 & 0 & * & * \\ 0 & 0 & 0 & 0 & * \end{pmatrix} = \begin{pmatrix} * & * & * & * & * \\ * & * & * & * & * \\ 0 & * & * & * & * \\ 0 & 0 & * & * & * \\ 0 & 0 & 0 & * & * \end{pmatrix},$$

т. е. $\mathbf{Q}_0$ является почти треугольной. Тогда

$$\mathbf{A}^{(1)} = \mathbf{R}_0\mathbf{Q}_0 = \begin{pmatrix} * & * & * & * & * \\ 0 & * & * & * & * \\ 0 & 0 & * & * & * \\ 0 & 0 & 0 & * & * \\ 0 & 0 & 0 & 0 & * \end{pmatrix} \begin{pmatrix} * & * & * & * & * \\ * & * & * & * & * \\ 0 & * & * & * & * \\ 0 & 0 & * & * & * \\ 0 & 0 & 0 & * & * \end{pmatrix} = \begin{pmatrix} * & * & * & * & * \\ * & * & * & * & * \\ 0 & * & * & * & * \\ 0 & 0 & * & * & * \\ 0 & 0 & 0 & * & * \end{pmatrix}.$$

Пример 6.10. Для матрицы $\mathbf{A} = \begin{pmatrix} 3 & -1 \\ -1 & 3 \end{pmatrix}$ QR-метод на первых четырех итерациях дает следующую последовательность матриц

$$\begin{pmatrix} 3,600 & 0,800 \\ 0,800 & 2,400 \end{pmatrix}, \begin{pmatrix} 3,882 & -0,471 \\ -0,471 & 2,118 \end{pmatrix}, \begin{pmatrix} 3,969 & 0,246 \\ 0,246 & 2,031 \end{pmatrix}, \begin{pmatrix} 3,992 & -0,125 \\ -0,125 & 2,008 \end{pmatrix}.$$

Внедиагональные элементы убывают как геометрическая прогрессия со знаменателем $\approx 0,5$ и после 15 итераций становятся меньше, чем 3×10^{-5}. Последовательность матриц $\{\mathbf{A}^{(k)}\}$ сходится к диагональной матрице с элементами 4 и 2 на диагонали.

При достаточно общих предположениях относительно исходной матрицы $\mathbf{A}$ последовательность матриц $\mathbf{A}^{(k)}$ в QR-алгоритме сходится к верхней треугольной или верхней блочно-треугольной (с блоками 2×2) матрице. Однако эта сходимость довольно медленная. QR-алгоритм может не сходиться совсем, если нарушены условия теоремы 2.4.

Пример 6.11. Для ортогональной матрицы $\mathbf{A} = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix}$ имеем $\mathbf{A} = \mathbf{QR}$, где $\mathbf{Q} = \mathbf{A}$ и $\mathbf{R} = \mathbf{E}$. Следовательно, QR-алгоритм дает последовательность $\mathbf{A}^{(k)} = \mathbf{A}$ и нет сходимости к треугольной или диагональной матрице. Так как матрица $\mathbf{A}$ имеет собственные значения $\lambda_{1,2} = \pm 1$, то здесь нарушены условия теоремы 2.4.

Для преодоления таких затруднений и ускорения сходимости QR-алгоритма следует применить сдвиги. Используются сдвиги двух видов. Пусть после m шагов QR-алгоритма матрица $\mathbf{A}^{(m)}$ в позиции (n, n) имеет число α_m. Тогда в качестве сдвига σ_m берем 1) $\sigma_m = \alpha_m$; или 2) σ_m – ближайшее к α_m собственное значение 2×2 матрицы, окаймляющей элемент α_m, т. е. занимающей правый нижний угол матрицы $\mathbf{A}^{(m)}$. Второй вариант, называемый *неявным QR-методом*, обычно предпочтительнее.

Рассмотрим последовательность матриц:

$$\mathbf{A}^{(k)} - \sigma_m\mathbf{E} = \mathbf{Q}_k\mathbf{R}_k, \quad \mathbf{A}^{(k+1)} = \mathbf{R}_k\mathbf{Q}_k + \sigma_m\mathbf{E}, \quad k = m, m+1, \ldots$$

задающих QR-алгоритм со сдвигом. Собственные значения матриц $\mathbf{A}^{(k)}$ и $\mathbf{A}^{(k+1)}$ совпадают, поскольку здесь, как и в обычном QR-алгоритме, эти матрицы подобны

$$\mathbf{A}^{(k+1)} = \mathbf{R}_k\mathbf{Q}_k + \sigma_m\mathbf{E} = \mathbf{Q}_k^{-1}(\mathbf{Q}_k\mathbf{R}_k + \sigma_m\mathbf{E})\mathbf{Q}_k = \mathbf{Q}_k^{-1}\mathbf{A}^{(k)}\mathbf{Q}_k.$$

Можно утверждать [16], что через m_1 шагов QR-алгоритма со сдвигом в позиции (n,n) будет находиться достаточно хорошее приближение к наименьшему по модулю собственному значению матрицы $\mathbf{A}$. Затем обычный QR-алгоритм применяется к матрице порядка $n-1$, получающейся из $\mathbf{A}^{(m+m_1)}$ вычеркиванием n-й строки и n-го столбца и т. д.

В примере 6.11 $\alpha_m = 0$, и матрица имеет собственные значения $\lambda = \pm 1$. В качестве сдвига для этой матрицы можно взять, например, $\sigma_m = 1$. Условия теоремы 6.4 будут выполнены, и QR-алгоритм быстро сходится.

Пусть $\mathbf{A}^{(\infty)}$ – предельная матрица QR-алгоритма. Как уже отмечалось, такая матрица будет верхней треугольной или верхней блочно-треугольной. Если все собственные значения матрицы $\mathbf{A}$ различны по модулю, то $\mathbf{A}^{(\infty)}$ – верхняя треугольная. Комплексно-сопряженным собственным значениям матрицы $\mathbf{A}$ в $\mathbf{A}^{(\infty)}$ соответствуют диагональные блоки 2×2. Если, наконец, собственное значение матрицы $\mathbf{A}$ имеет кратность p, то ему в $\mathbf{A}^{(\infty)}$ соответствует диагональный блок порядка p. Таким образом, если матрица не имеет кратных собственных значений, то QR-алгоритм сводит решение полной проблемы собственных значений к нахождению собственных значений диагональных блоков размерности не выше двух.

Пример 6.12. Рассмотрим матрицу $\mathbf{A} = \begin{pmatrix} 5 & 1 & 0 \\ 1 & 5 & 2 \\ 0 & 2 & 5 \end{pmatrix}$ с собственными значениями: $\lambda_1 = 5$; $\lambda_2 = (7+\sqrt{29})/2 \approx 6,1926$; $\lambda_3 = (7-\sqrt{29})/2 \approx 0,8074$.

Применим неявный QR-метод со сдвигами. Проведем три итерации со сдвигами $\sigma_0 = 6$, $\sigma_1 = 0,1861$, $\sigma_2 = 0,0065$. Матрицы $\mathbf{A}^{(k)} - \sigma_k \mathbf{E}$, $k = 0,1,2$, последовательно преобразуем в подобные матрицы

$$\mathbf{A}^{(1)} = \begin{pmatrix} -3,5000 & 2,0616 & 0,0000 \\ 2,0616 & -2,6765 & 0,1664 \\ 0,0000 & 0,1664 & 0,1765 \end{pmatrix},$$

$$\mathbf{A}^{(2)} = \begin{pmatrix} -5,2464 & 0,7329 & 0,0000 \\ 0,7329 & -1,3184 & 0,0072 \\ 0,0000 & 0,0072 & 0,0064 \end{pmatrix},$$

$$\mathbf{A}^{(3)} = \begin{pmatrix} -5,3785 & 0,1673 & 0,0000 \\ 0,1673 & -1,1993 & 0,0000 \\ 0,0000 & 0,0000 & 0,0000 \end{pmatrix}.$$

Найдено собственное значение $\lambda_2 = \sigma_0 + \sigma_1 + \sigma_2 = 6,1926$. Применяя теперь QR-метод к 2×2 матрице $(\mathbf{B}^{(3)} - \sigma_3 \mathbf{E})$ со сдвигом $\sigma_3 = -1,1926$, получаем

$$\mathbf{B}^{(3)} = \begin{pmatrix} -5,3785 & 0,1673 \\ 0,1673 & -1,1993 \end{pmatrix} \to \mathbf{B}^{(4)} = \begin{pmatrix} -4,1926 & 0,0000 \\ 0,0000 & 0,0000 \end{pmatrix}.$$

Отсюда $\lambda_1 = \lambda_2 + \sigma_3 = 5$, $\lambda_3 = \lambda_1 - 4,1926 = 0,8074$.

§ 6.9. Метод Ланцоша

Приведение симметрической матрицы к трехдиагональной форме дает существенные вычислительные преимущества при дальнейшем поиске ее собственных значений. Такое преобразование может быть проведено путем многократного применения матриц вращения или отражения. Это требует однако большого объема вычислений. Гораздо более эффективным является использование явных формул для нужного преобразования. Такой подход известен как *метод трехдиагонализации Ланцоша*.

Пусть требуется преобразовать симметрическую матрицу $\mathbf{A}$ к трехдиагональной матрице $\mathbf{T}$, используя преобразование подобия $\mathbf{Q}^T\mathbf{A}\mathbf{Q} = \mathbf{T}$, где $\mathbf{Q}$ – ортогональная матрица. Тогда

$$\mathbf{A}\mathbf{Q} = \mathbf{Q}\mathbf{T}. \tag{6.6}$$

Обозначим i-й столбец матрицы $\mathbf{Q}$ через $\mathbf{q}_i$ и пусть $\mathbf{T}$ имеет вид

$$\mathbf{T} = \begin{pmatrix} a_1 & b_2 & 0 & \dots & \dots & 0 \\ b_2 & a_2 & b_3 & \dots & \dots & 0 \\ 0 & \ddots & \ddots & \ddots & & \vdots \\ \vdots & & \ddots & \ddots & \ddots & 0 \\ 0 & \dots & \dots & b_{n-1} & a_{n-1} & b_n \\ 0 & \dots & \dots & 0 & b_n & a_n \end{pmatrix}.$$

Равенство (6.6) можно записать в виде

$$\mathbf{A}\mathbf{q}_i = b_i\mathbf{q}_{i-1} + a_i\mathbf{q}_i + b_{i+1}\mathbf{q}_{i+1}, \quad i = 1, 2, \dots, n, \tag{6.7}$$

где $\mathbf{q}_0 = \mathbf{q}_{n+1} = \mathbf{0}$ и $b_1 = b_{n+1} = 0$. Поскольку $\mathbf{Q}$ – ортогональная матрица, то получаем

$$\mathbf{q}_i^T\mathbf{q}_{i-1} = \mathbf{q}_i^T\mathbf{q}_{i+1} = 0 \quad \text{и} \quad \mathbf{q}_i^T\mathbf{q}_i = 1.$$

Умножая равенство (6.7) на $\mathbf{q}_i^T$, находим

$$a_i = \mathbf{q}_i^T\mathbf{A}\mathbf{q}_i. \tag{6.8}$$

Равенство (6.7) можно переписать в виде

$$b_{i+1}\mathbf{q}_{i+1} = \mathbf{A}\mathbf{q}_i - b_i\mathbf{q}_{i-1} - a_i\mathbf{q}_i = \mathbf{r}_i, \tag{6.9}$$

откуда

$$b_{i+1} = \pm\|\mathbf{r}_i\|_2. \tag{6.10}$$

Задавая каким-либо образом $\mathbf{q}_1$ и используя (6.8) – (6.10), заданную симметрическую матрицу можно привести к трехдиагональной форме.

Пример 6.13. Рассмотрим трехдиагонализацию матрицы

$$\mathbf{A} = \begin{pmatrix} 2 & 3 & 4 \\ 3 & 1 & -1 \\ 4 & -1 & 2 \end{pmatrix}.$$

Положим $\mathbf{q}_1 = (1;0;0)^T$ и, пользуясь формулами

$$a_i = \mathbf{q}_i^T \mathbf{A}\mathbf{q}_i, \quad \mathbf{r}_i = (\mathbf{A} - a_i\mathbf{E})\mathbf{q}_i - b_i\mathbf{q}_{i-1}, \quad b_{i+1} = \|\mathbf{r}_i\|_2, \quad \mathbf{q}_{i+1} = \frac{1}{b_{i+1}}\mathbf{r}_i,$$

проведем вычисления по методу Ланцоша:

$$a_1 = \mathbf{q}_1^T\mathbf{A}\mathbf{q}_1 = 2; \qquad \mathbf{r}_1 = (\mathbf{A} - a_1\mathbf{E})\mathbf{q}_1 = (0;3;4)^T;$$
$$b_2 = \|\mathbf{r}_1\|_2 = 5; \qquad \mathbf{q}_2 = \frac{1}{b_2}\mathbf{r}_1 = (0;0,6;0,8)^T;$$
$$a_2 = \mathbf{q}_2^T\mathbf{A}\mathbf{q}_2 = 0,68; \quad \mathbf{r}_2 = (\mathbf{A} - a_2\mathbf{E})\mathbf{q}_2 - b_2\mathbf{q}_1 = \frac{1}{5}(0;-3,04;2,28)^T;$$
$$b_3 = \|\mathbf{r}_2\|_2 = 0,76; \quad \mathbf{q}_3 = \frac{1}{b_3}\mathbf{r}_2 = (0;-0,8;0,6)^T; \quad a_3 = \mathbf{q}_3^T\mathbf{A}\mathbf{q}_3 = 2,32.$$

Таким образом,

$$\mathbf{Q} = (\mathbf{q}_1, \mathbf{q}_2, \mathbf{q}_3) = \begin{pmatrix} 1,0 & 0,0 & 0,0 \\ 0,0 & 0,6 & -0,8 \\ 0,0 & 0,8 & 0,6 \end{pmatrix},$$

$$\mathbf{T} = \begin{pmatrix} a_1 & b_2 & 0 \\ b_2 & a_2 & b_3 \\ 0 & b_3 & a_3 \end{pmatrix} = \begin{pmatrix} 2,00 & 5,00 & 0,00 \\ 5,00 & 0,68 & 0,76 \\ 0,00 & 0,76 & 2,32 \end{pmatrix}.$$

Здесь $\mathbf{T} = \mathbf{Q}^T\mathbf{A}\mathbf{Q}$ и легко проверить, что ту же самую матрицу получим, применяя матрицы отражения.

Метод Ланцоша предпочтителен по сравнению с применением матриц отражения. Это особенно существенно при работе с разреженными матрицами, так как при использовании метода Ланцоша не нужно формировать произведения вида $\mathbf{HAH}$, где $\mathbf{H}$ – матрица отражения. После трехдиагонализации для определения собственных значений может быть использован QR-алгоритм или, например, метод бисекций [16]. Собственные векторы можно затем получить применением обратного степенного метода.

Отметим, что метод Ланцоша является частным случаем метода Арнольди приведения несимметрической матрицы к верхней форме Хессенберга, который также базируется на преобразовании подобия $\mathbf{H} = \mathbf{Q}^T\mathbf{A}\mathbf{Q}$, где $\mathbf{Q}$ – ортогональная матрица. Метод Арнольди реализуется по явным формулам и эффективнее использования матриц отражения. Подробное изложение этого метода можно найти, например, в [10].

§ 6.10. Сингулярное разложение

При решении переопределенных линейных систем, когда число уравнений m может превышать число неизвестных n ($m \geq n$) важную роль играет *сингулярное разложение* прямоугольных матриц.

Теорема 6.5. *Пусть $m \times n$ матрица $\mathbf{A}$ имеет ранг r. Тогда существуют $m \times m$ ортогональная матрица $\mathbf{U}$, $n \times n$ ортогональная матрица $\mathbf{V}$ и $m \times n$ «диагональная» матрица $\mathbf{\Sigma}$ такие, что*

$$\mathbf{U}^T\mathbf{A}\mathbf{V} = \mathbf{\Sigma} = \begin{pmatrix} \sigma_1 & 0 & \dots & 0 & \dots & 0 \\ 0 & \sigma_2 & \dots & 0 & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & \sigma_r & \dots & 0 \\ 0 & 0 & \dots & 0 & \dots & 0 \\ \vdots & \vdots & \dots & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & 0 & \dots & 0 \end{pmatrix}.$$

Числа $\sigma_1, \sigma_2, \dots, \sigma_r$ называются сингулярными значениями матрицы $\mathbf{A}$. Все они вещественны и положительны и могут быть упорядочены таким образом, что

$$\sigma_1 \geq \sigma_2 \geq \dots \geq \sigma_r > 0.$$

Столбцы матриц $\mathbf{U}$ и $\mathbf{V}$ называются левыми и правыми сингулярными векторами. Представление матрицы $\mathbf{A}$ в виде $\mathbf{A} = \mathbf{U}\mathbf{\Sigma}\mathbf{V}^T$ называется сингулярным разложением матрицы $\mathbf{A}$.

Отметим, что ненулевые собственные значения матриц $\mathbf{A}\mathbf{A}^T$ и $\mathbf{A}^T\mathbf{A}$ одни и те же, а сингулярные числа $\mathbf{A}$ – квадратные корни из этих собственных значений. Кроме того, левые и правые сингулярные векторы суть частные выборы собственных векторов для $\mathbf{A}\mathbf{A}^T$ и $\mathbf{A}^T\mathbf{A}$.

Пусть решается система

$$\mathbf{A}\mathbf{x} = \mathbf{b},$$

где $\mathbf{A}$ – заданная $m \times n$ матрица ($m \geq n$), а вектор правой части $\mathbf{b}$ имеет длину m. Отметим, что сюда попадает важный на практике случай, когда $\mathbf{A}$ – квадратная вырожденная матрица. Используя сингулярное разложение матрицы $\mathbf{A}$, нашу линейную систему можно переписать в виде

$$\mathbf{U}\mathbf{\Sigma}\mathbf{V}^T\mathbf{x} = \mathbf{b} \quad \text{или} \quad \mathbf{\Sigma}\mathbf{z} = \mathbf{d},$$

где $\mathbf{z} = \mathbf{V}^T\mathbf{x}$ и $\mathbf{d} = \mathbf{U}^T\mathbf{b}$. Полученная система уравнений является диагональной и, следовательно, может быть легко решена. Подробное изложение метода сингулярного разложения можно найти, например, в [8].

§ 6.11. Задачи

6.1. Пользуясь теоремой Гершгорина, локализовать собственные значения следующих матриц:

$$\text{а) } \mathbf{A} = \begin{pmatrix} 1 & 0 & -2 \\ 0 & 0 & 3 \\ 1 & 0 & 3 \end{pmatrix}; \quad \text{б) } \mathbf{A} = \begin{pmatrix} -1 & 2 & 0 \\ 4 & 4 & 0 \\ 0 & 0 & 1 \end{pmatrix}; \quad \text{в) } \mathbf{A} = \begin{pmatrix} -6 & 1 & 1 \\ -1 & 0 & 1 \\ 0 & 0 & 3 \end{pmatrix}.$$

6.2. Пусть задана вещественная трехдиагональная матрица

$$\mathbf{A} = \begin{pmatrix} b_1 & c_1 & 0 & \dots & 0 & 0 \\ a_2 & b_2 & c_2 & \dots & 0 & 0 \\ 0 & a_3 & b_3 & \dots & 0 & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & 0 & \dots & b_{n-1} & c_{n-1} \\ 0 & 0 & 0 & \dots & a_n & b_n \end{pmatrix}.$$

Доказать, что:

а) все собственные значения матрицы $\mathbf{A}$ вещественны, если $a_{i+1}c_i > 0$ для $i = 1, 2, \dots, n-1$;

б) $|\lambda_k(\mathbf{A})| < 1$ для всех k, если $|a_i| + |b_i| + |c_i| \le 1$ для всех i при $a_1 = c_n = 0$ и если хотя бы для одного значения индекса i неравенство строгое, а $a_{i+1}c_i \neq 0$, $i = 1, 2, \dots, n-1$.

6.3. Доказать положительную определенность матрицы

$$\mathbf{A} = \begin{pmatrix} 12 & -6 & 3 & -2 \\ -6 & 18 & -6 & 6 \\ 3 & -6 & 24 & 15 \\ -2 & 6 & 15 & 30 \end{pmatrix}.$$

6.4. Пусть $n \times n$ матрица $\mathbf{A}$ имеет диагональное преобладание, причем $a_{ii} > \sum_{j \neq i} |a_{ij}|$ и $a_{ij} < 0$ для $j \neq i$. Доказать, что из $\mathbf{Ax} \ge 0$ следует $\mathbf{x} \ge 0$ и из $\mathbf{Ax} \le 0$ следует $\mathbf{x} \le 0$ (знаки $\le$ и $\ge$ означают, что эти неравенства имеют место для всех компонент). Покажите, что элементы матрицы $\mathbf{A}^{-1}$ неотрицательны.

6.5. Матрица $\mathbf{A} = \begin{pmatrix} 101 & -90 \\ 110 & -98 \end{pmatrix}$ имеет собственные значения $\lambda_1 = 1$, $\lambda_2 = 2$. Возмущенная матрица $\mathbf{A}+\Delta\mathbf{A} = \begin{pmatrix} 101-\varepsilon & -90-\varepsilon \\ 110 & -98 \end{pmatrix}$. Используя теорему Бауера-Файка и норму $\|\cdot\|_1$, оценить изменение собственных значений. Найти собственные значения матрицы $\mathbf{A} + \Delta\mathbf{A}$ для $\varepsilon = 0,001$ и сравнить их с полученной оценкой.

6.6. Матрица $\mathbf{A} = \begin{pmatrix} -5,2 & 1,0 & 0,0 \\ 4,0 & 1,1 & -4,0 \\ -10,2 & 0,9 & 5,0 \end{pmatrix}$ имеет собственные значения $\lambda_1 = 1$, $\lambda_2 = 5,1$, $\lambda_3 = -5,2$. Степенной метод с некоторым начальным приближением

$\mathbf{x}^{(0)}$ не сходится даже после 500 итераций. Тот же метод для матрицы $\mathbf{B} = \mathbf{A} - 4\mathbf{E}$ сходится очень быстро. Объясните это явление.

6.7. Использовать степенной метод для нахождения наибольшего по модулю собственного значения и соответствующего собственного вектора симметрической матрицы $\mathbf{A} = \begin{pmatrix} 5 & -1 & -1 & -1 \\ -1 & 5 & -1 & -1 \\ -1 & -1 & 5 & -1 \\ -1 & -1 & -1 & 5 \end{pmatrix}$. В качестве начального приближения взять: а) $\mathbf{x}^{(0)} = (1;1;1;1)^T$; б) $\mathbf{x}^{(0)} = (0;0;0;1)^T$. Почему это приводит к разным результатам?

6.8. Применяя обратный степенной метод, найти наименьшее собственное значение матрицы из задачи 6.7.

6.9. Для матрицы $\mathbf{A} = \begin{pmatrix} 2 & -1 \\ -1 & 2 \end{pmatrix}$, используя степенной метод с начальными векторами $\mathbf{u}_0^{(1)} = (2;0)^T$; $\mathbf{u}_0^{(2)} = (1;1)^T$; $\mathbf{u}_0^{(3)} = (1+\varepsilon, 1)^T$ найти $\mathbf{A}^5\mathbf{u}_0^{(i)}$, $i = 1,2,3$. Объяснить полученные результаты.

6.10. Матрица $\mathbf{A} = \begin{pmatrix} 4 & -1 \\ -1 & 4 \end{pmatrix}$ имеет собственные значения $\lambda_1 = 3$ и $\lambda_2 = 5$. Взяв в качестве начального приближения вектор $\mathbf{u}_0 = (1;0)^T$, проделать три итерации:

а) степенного метода $\mathbf{u}_{k+1} = \mathbf{A}\mathbf{u}_k$;

б) обратного степенного метода $\mathbf{A}\mathbf{u}_{k+1} = \mathbf{u}_k$;

в) обратного степенного метода со сдвигом $(\mathbf{A} - \alpha\mathbf{E})\mathbf{u}_{k+1} = \mathbf{u}_k$, где величина сдвига $\alpha = (\mathbf{u}_0, \mathbf{A}\mathbf{u}_0)/\|\mathbf{u}_0\|^2$. Сравнить результаты.

6.11. Методом исчерпывания при $p = 4$ найти собственные значения и собственные векторы матрицы $\mathbf{A} = \begin{pmatrix} p & 1 & 0 & 0 \\ 1 & p & 1 & 0 \\ 0 & 1 & p & 1 \\ 0 & 0 & 1 & p \end{pmatrix}$.

6.12. Методом вращений Якоби найти собственные значения и собственные векторы матрицы $\mathbf{A} = \begin{pmatrix} 1 & 1 & 3 \\ 1 & 5 & 1 \\ 3 & 1 & 1 \end{pmatrix}$.

6.13. Методом вращений Якоби при $p = 4$ найти собственные значения и собственные векторы матрицы $\mathbf{A} = \begin{pmatrix} p & 1 & 1 & 1 \\ 1 & p & 1 & 1 \\ 1 & 1 & p & 1 \\ 1 & 1 & 1 & 1 \end{pmatrix}$.

6.14. Методом вращений Гивенса получить QR-разложение матрицы

$$\mathbf{A} = \begin{pmatrix} 180 & 24 & 190 \\ 240 & 157 & 45 \\ 225 & 230 & 425 \end{pmatrix}.$$

6.15. Пусть $\mathbf{Q}_1$ и $\mathbf{Q}_2$ – ортогональные матрицы. Показать, что $\mathbf{Q}_1\mathbf{Q}_2$ – ортогональная матрица.

6.16. Используя метод отражений, найти QR-разложения матриц:

$$\text{а) } \mathbf{A} = \begin{pmatrix} 2 & -4 & 0 \\ 2 & 1 & 0 \\ 1 & 3 & 15 \end{pmatrix}; \qquad \text{б) } \mathbf{A} = \begin{pmatrix} 1 & 1 & 2 \\ 2 & -1 & -1 \\ 2 & -4 & 5 \end{pmatrix}.$$

6.17. Показать, что в матрице $\mathbf{A} = \begin{pmatrix} \alpha & \gamma \\ \gamma & \beta \end{pmatrix}$ ближайшее к элементу β собственное значение находится по формуле

$$\lambda = \beta - \frac{\gamma^2}{\delta + \operatorname{sign}(\delta)\sqrt{\delta^2 + \gamma^2}}, \quad \delta = \frac{\alpha - \beta}{2}.$$

6.18. Начав с вектора $\mathbf{q}_1 = (0, 1, 0)$, методом Ланцоша привести к трехдиагональной форме матрицу $\mathbf{A} = \begin{pmatrix} 2 & 3 & 4 \\ 3 & 1 & -1 \\ 4 & -1 & 2 \end{pmatrix}$.

6.19. Показать, что для максимального и минимального собственных значений симметрической матрицы $\mathbf{A} = (a_{ij})$ справедливы оценки

$$\lambda_{\min}(\mathbf{A}) \le \min_{1 \le i \le n} a_{ii}; \quad \lambda_{\max}(\mathbf{A}) \ge \max_{1 \le i \le n} a_{ii}.$$

6.20. Доказать, что если $\mathbf{A}$ – симметрическая и положительно определенная матрица, а $\mathbf{B}$ – симметрическая матрица, то система собственных векторов матрицы $\mathbf{AB}$ полна.

6.21. Пусть $\mathbf{A}$ – симметризуемая матрица, т. е. существует невырожденная матрица $\mathbf{Q}$ такая, что $\mathbf{QAQ}^{-1}$ – симметрическая матрица. Доказать, что система собственных векторов матрицы $\mathbf{A}$ полна.

6.22. Доказать, что если $\mathbf{A}, \mathbf{B}$ – симметрические, положительно определенные и коммутирующие матрицы, то матрица $\mathbf{AB}$ положительно определена.

6.23. Пусть дана вещественная $n \times n$ матрица

$$\mathbf{A}(a, b) = \begin{pmatrix} a & b & 0 & 0 & \dots & 0 \\ b & a & b & 0 & \dots & 0 \\ 0 & b & a & b & \dots & 0 \\ \vdots & \vdots & \ddots & \ddots & \ddots & \vdots \\ 0 & 0 & \dots & b & a & b \\ 0 & 0 & \dots & 0 & b & a \end{pmatrix}.$$

а) для матрицы $\mathbf{A}(a,b)$ найти все собственные значения и собственные векторы;

б) доказать, что матрица $\mathbf{A}(a,b)$ положительно определена тогда и только тогда, когда $a - 2|b|\cos\dfrac{\pi}{n+1} > 0$.

Ответы, указания, решения

Глава 1

1.1. Полагая

$$\varphi(x) = x - f(x)\frac{x-a}{f(x)-f(a)},$$

метод хорд можно переписать как метод простой итерации $x_{k+1} = \varphi(x_k)$. Вблизи корня x^* уравнения $f(x) = 0$ можно записать

$$\varphi(x) = \varphi(x^*) + \varphi'(x^*)(x-x^*) + \varphi''(x^* + \theta(x-x^*))\frac{(x-x^*)^2}{2}, \quad 0 < \theta < 1,$$

откуда, полагая $x = x_k$, находим

$$x_{k+1} = x^* + \varphi'(x^*)(x_k - x^*) + \varphi''(x^* + \theta(x_k - x^*))(x_k - x^*)^2/2.$$

Так как

$$\begin{aligned}\varphi'(x^*) &= 1 + \frac{f'(x^*)}{f(a)}(x^*-a) = \frac{1}{f(a)}\Big[f(x^*) + f'(x^*)(a-x^*) + \\ &\quad + f''(\xi)\frac{(a-x^*)^2}{2} + f'(x^*)(x^*-a)\Big] = \frac{f''(\xi)}{f(a)}\frac{(a-x^*)^2}{2}, \quad \xi \in [a, x^*],\end{aligned}$$

то, выбирая x_k достаточно близким к x^*, можно обеспечить $|\varphi'(x^*)| < 1$ и получить линейную скорость сходимости метода хорд.

1.2. Метод имеет линейную скорость сходимости.

1.3. Пусть $\varphi(x) = x - f(x)/f'(x)$. Тогда в окрестности корня x^* имеем

$$\varphi(x) = \varphi(x^*) + \varphi'(x^*)(x-x^*) + \varphi''(\xi)(x-x^*)^2/2.$$

Полагая здесь $x = x_k$, получаем

$$x_{k+1} = x^* + \varphi'(x^*)(x_k - x^*) + \varphi''(\xi)(x_k - x^*)^2/2.$$

Вблизи корня x^* имеем $f(x) \approx C(x-x^*)^m$, где $C = const$, и

$$\varphi'(x) = \frac{f(x)f''(x)}{[f'(x)]^2} \approx \frac{C(x-x^*)^m Cm(m-1)(x-x^*)^{m-2}}{C^2m^2(x-x^*)^{2m-2}} = \frac{m-1}{m}.$$

Поэтому

$$\frac{|x_{k+1}-x^*|}{|x_k-x^*|}\approx\frac{m-1}{m}.$$

В частности, отсюда следует, что чем выше кратность корня x^*, тем медленнее будет сходимость.

1.4. Запишем интерполяционный многочлен в форме Ньютона (см. гл. 2)

$$L_2(x)=f(x_k)+(x-x_k)f[x_k,x_{k-1}]+(x-x_k)(x-x_{k-1})f[x_k,x_{k-1},x_{k-2}].$$

Приравнивая его нулю, получаем квадратное уравнение $aw^2+bw+c=0$, где $w=x-x_k$, $a=f[x_k,x_{k-1},x_{k-2}]$, $b=a(x_k-x_{k-1})+f[x_k,x_{k-1}]$, $c=f(x_k)$. Тот из двух корней квадратного уравнения, который меньше по модулю, определяет новое приближение $x_{k+1}=x_k+w$. Для начала счета выбирают какие-то три первых приближения x_0, x_1, x_2. Рассуждая аналогично тому, как это делалось в методе секущих, можно показать, что вблизи простого корня выполняется соотношение

$$|x_{k+1}-x^*|\approx\left|\frac{f'''(x^*)}{6f'(x^*)}\right|^{0,42}|x_k-x^*|^{1,84}.$$

Для кратных корней этот метод имеет скорость сходимости $p=1,23$.

1.5. а) $|\varphi'(x)|=\cos^{-2}(x)\geq 1$ – метод расходится.

б) $|\varphi'(x)|=[1+(1+x)^2]^{-1}\leq 1$ – метод сходится.

Следует выбрать второе выражение.

1.6. Пусть x^* – точное решение уравнения $f(x)=0$. Тогда

$$\begin{aligned} w_{k+1} &= x_{k+1}-x^*=x_k-\frac{f(x_k)}{f'(x_k)}-x^*=\frac{w_kf'(x_k)-f(x_k)}{f'(x_k)},\\ 0 &= f(x^*)=f(x_k-w_k)=f(x_k)-w_kf'(x_k)+\frac{1}{2}f''(\xi_k)w_k^2. \end{aligned}$$

Отсюда

$$w_{k+1}=\frac{1}{2}\frac{f''(\xi_k)}{f'(x_k)}w_k^2.$$

Так как $f'(x)f''(x)>0$, то итерационная последовательность, построенная по методу Ньютона, монотонно убывает и сходится к корню x^*.

1.7. $x_{k+1}=\frac{1}{2}\Big(x_k+\frac{9}{x_k}\Big)$, $k=0,1,\ldots,$. Метод сходится, если $x_0\neq 0$.

1.8. Полагая $\varphi(x)=x-\tau f(x)$, имеем

$$x_{k+1}-x^*=\varphi'(x^*)(x_k-x^*)+\varphi''(\xi)(x_k-x^*)/2,\quad \xi\in[x_k,x^*].$$

Так как $\min_\tau|\varphi'(x^*)|$ достигаетсся при $1-\tau m=-(1-\tau M)$, то отсюда получаем $\tau_{opt}=2/(m+M)$. В этом случае $|\varphi'(x^*)|<1$ и метод релаксации имеет линейную скорость сходимости.

1.9. $x = 16,21$.

1.10. а) Положим $f_1(x,y) = \sin(x+1) - y$, $f_2(x,y) = x^2 + y^2 - 1$. Тогда расчетные формулы метода Ньютона принимают вид:

$$\begin{aligned} x_{k+1} &= x_k - J_k^{-1}[2y_k f_1(x_k,y_k) + f_2(x_k,y_k)], \\ y_{k+1} &= y_k - J_k^{-1}[\cos(x_k+1) f_2(x_k,y_k) - 2x_k f_1(x_k,y_k)], \\ J_k &= 2[x_k + y_k + y_k\cos(x_k+1)]. \end{aligned}$$

б) Полагая $f_1(x,y) = (x-0,5)^2 + 0,5 - y$, $f_2(x,y) = x^2 + y^2 - 1$, получаем

$$\begin{aligned} x_{k+1} &= x_k - J_k^{-1}[2y_k f_1(x_k,y_k) + f_2(x_k,y_k)], \\ y_{k+1} &= y_k - J_k^{-1}[(x_k-0,5) f_2(x_k,y_k) - 2x_k f_1(x_k,y_k)], \\ J_k &= 2(x_k - y_k + 2x_k y_k). \end{aligned}$$

1.11. Пусть (x^*, y^*) – точное решение нашей системы. Тогда по формуле Тейлора для случая двух переменных получаем

$$\begin{aligned} x^* - x_{k+1} &= \frac{\partial\varphi_1(\xi_{1,k},\eta_{1,k})}{\partial x}(x^* - x_k) + \frac{\partial\varphi_1(\xi_{1,k},\eta_{1,k})}{\partial y}(y^* - y_k), \\ y^* - y_{k+1} &= \frac{\partial\varphi_2(\xi_{2,k},\eta_{2,k})}{\partial x}(x^* - x_k) + \frac{\partial\varphi_2(\xi_{2,k},\eta_{2,k})}{\partial y}(y^* - y_k), \end{aligned}$$

где точки $(\xi_{1,k},\eta_{1,k})$ и $(\xi_{2,k},\eta_{2,k})$ лежат на отрезке, соединяющем точки (x^*,y^*) и (x_k,y_k).

В матричной форме эти уравнения принимают вид

$$\begin{pmatrix} x^* - x_{k+1} \\ y^* - y_{k+1} \end{pmatrix} = \begin{pmatrix} \dfrac{\partial\varphi_1(\xi_{1,k},\eta_{1,k})}{\partial x} & \dfrac{\partial\varphi_1(\xi_{1,k},\eta_{1,k})}{\partial y} \\ \dfrac{\partial\varphi_2(\xi_{2,k},\eta_{2,k})}{\partial x} & \dfrac{\partial\varphi_2(\xi_{2,k},\eta_{2,k})}{\partial y} \end{pmatrix} \begin{pmatrix} x^* - x_k \\ y^* - y_k \end{pmatrix}.$$

Отсюда

$$|x^* - x_{k+1}| + |y^* - y_{k+1}| \le \left[\left|\frac{\partial\varphi_1}{\partial x}\right| + \left|\frac{\partial\varphi_2}{\partial x}\right|\right]|x^* - x_k| + \left[\left|\frac{\partial\varphi_1}{\partial y}\right| + \left|\frac{\partial\varphi_2}{\partial y}\right|\right]|y^* - y_k|.$$

Если теперь

$$\left|\frac{\partial\varphi_1}{\partial x}\right| + \left|\frac{\partial\varphi_2}{\partial x}\right| < 1 \quad \text{и} \quad \left|\frac{\partial\varphi_1}{\partial y}\right| + \left|\frac{\partial\varphi_2}{\partial y}\right| < 1,$$

то

$$|x^* - x_{k+1}| + |y^* - y_{k+1}| < |x^* - x_k| + |y^* - y_k|.$$

Глава 2

2.1. По определению разделенных разностей получаем

$$\begin{aligned} f[x_i, x_{i+1}, \ldots, x_{i+n+1}] &= \frac{f[x_{i+1}, \ldots, x_{i+n+1}] - f[x_i, \ldots, x_{i+n}]}{x_{i+n+1} - x_i} = \\ &= \frac{0-0}{x_{i+n+1} - x_i} = 0. \end{aligned}$$

2.2. Запишем два соседних многочлена Лагранжа в форме Ньютона

$$\begin{aligned} L_{i,n+1}(x) &= f_{i+1} + f[x_{i+1}, x_{i+2}] + \ldots + \\ &\quad + f[x_{i+1}, \ldots, x_{i+n}, x_{i+n+1}](x - x_{i+1}) \ldots (x - x_{i+n}), \\ L_{i,n+1}(x) &= f_{i+1} + f[x_{i+1}, x_{i+2}] + \ldots + \\ &\quad + f[x_{i+1}, \ldots, x_{i+n}, x_i](x - x_{i+1}) \ldots (x - x_{i+n}), \end{aligned}$$

Отсюда, учитывая свойства симметрии разделенных разностей, получаем

$$\begin{aligned} L_{i+1,n+1}(x) &- L_{i,n+1}(x) = \\ &= (f[x_{i+1}, \ldots, x_{i+n+1}] - f[x_i, \ldots, x_{i+n}])(x - x_{i+1}) \ldots (x - x_{i+n}) = \\ &= (x_{i+n+1} - x_i) f[x_i, \ldots, x_{i+n+1}](x - x_{i+1}) \ldots (x - x_{i+n}). \end{aligned}$$

2.3. Так как по условию задачи разделенные разности порядка $n+1$ равны нулю, то из задачи 2.2 следует, что

$$L_{i+1,n+1}(x) \equiv L_{i,n+1}(x), \quad i = 0, 1, \ldots, N-n-1.$$

Поэтому $L_{i,n+1}(x_j) = f_j$, $j = 0, 1, \ldots, N$.

2.4. Многочлен Лагранжа, интерполирующий данные (x_i, f_i), $i = 0, 1, \ldots, N$, принято записывать следующим образом

$$L_N(x) = \sum_{i=0}^{N} f_i l_i(x), \quad l_i(x) = \frac{\omega_N(x)}{(x - x_i)\omega'_N(x_i)}, \quad \omega_N(x) = \prod_{i=0}^{N} (x - x_i).$$

Так как $\sum_{i=0}^{N} l_i(x) \equiv 1$, эту формулу можно переписать в виде

$$L_N(x) = \frac{\sum_{i=0}^{N} f_i l_i(x)}{\sum_{i=0}^{N} l_i(x)} = \frac{\sum_{i=0}^{N} f_i [(x - x_i)\omega'_N(x_i)]^{-1}}{\sum_{i=0}^{N} [(x - x_i)\omega'_N(x_i)]^{-1}} = \frac{\sum_{i=0}^{N} w_i f_i/(x - x_i)}{\sum_{i=0}^{N} w_i/(x - x_i)},$$

где $w_i = 1/\omega'_N(x_i)$.

2.5. а) Таблица разделенных разностей:

x_i	f_i	$f[x_i, x_{i+1}]$	$f[x_i, x_{i+1}, x_{i+2}]$	$f[x_i, \ldots, x_{i+3}]$
0	1			
1	0	-1		
2	-1	-1	0	
3	0	1	1	$1/3$

б) $L_3(x) = 1 - x + x(x-1)(x-2)/3$.

в) $L_3(1,5) = -5/8, \quad L_3'(1,5) = -13/12$.

г) $f(1,5) - L_3(1,5) \approx -0,082, \quad f'(1,5) - L_3'(1,5) \approx -0,027$.

2.6. Для многочлена $P_4(x) = a_0 + a_1x + a_2x^2 + a_3x^3 + a_4x^4$ имеем

$$\Delta^2 P_4(x) = 2a_2 + 6(1+x)a_3 + (14 + 24x + 12x^2)a_4,$$
$$\Delta^3 P_4(x) = 6a_3 + (36 + 24x)a_4, \quad \Delta^4 P_5(x) = 24a_4.$$

Так как $\Delta^2 P_4(0) = 0$, $\Delta^3 P_4(0) = 6$, $\Delta^4 P_4(0) = 24$, то $a_4 = 1$, $a_3 = -5$ и $a_2 = 8$. Поэтому

$$\Delta^2 P_4(x) = -6x + 12x^2 \quad \text{и} \quad \Delta^2 P_4(10) = 1140.$$

2.7. См. доказательство теоремы 2.6.

2.8. Согласно оценке

$$E = |f(x) - L_1(x)| \le \frac{h^2}{8}\|f''\|_C = \frac{1}{8N^2}\|f''\|_C \le \varepsilon, \quad 0 \le x \le 1$$

получаем а) $E \le \pi^2/200$; б) $N \ge \pi/\sqrt{8\varepsilon}$.

2.9. Согласно оценке

$$|f'(x) - L_1'(x)| \le \frac{h}{2}M + \frac{2\varepsilon}{h} = \varphi(h,\varepsilon), \quad M = \|f''\|_C, \quad 0 \le x \le \pi$$

получаем $h = \pi/N = (\varepsilon/M)^{1/2} \approx 0,0707$ и $N = 45$.

2.10. Функцию B_3^L можно записать в виде

$$B_3^L(x) = \begin{cases} \dfrac{\omega_{k,3}(x)}{x\omega_{k,3}'(0)}, & k \le x \le k+1, \\ & k = -2, -1, 0, 1, \\ 0 & \text{в противном случае,} \end{cases} \qquad \omega_{k,3}(x) = \prod_{j=k-1}^{k+2}(x-j).$$

Отсюда следует, что функция B_3^L образована четырьмя кубическими фундаментальными многочленами Лагранжа и удовлетворяет следующим условиям

$B_3^L(\pm 2) = B_3^L(\pm 1) = 0$, $B_3^L(0) = 1$. Используя приведенную формулу, получаем

$$\sum_{k=-2}^{1} B_3^L(x+k+j) = \sum_{k=-2}^{1} \frac{\omega_{k,3}(x+k+j)}{(x+k+j)\omega'_{k,3}(k+j)} \equiv 1, \quad j \le x \le j+1.$$

Таким образом, функции B_3^L образуют разбиение единицы на всей оси. Доказательство линейной независимости функций B_3^L аналогично доказательству теоремы 2.6.

2.11. Согласно оценке

$$E = |f(x) - L_{3,i}(x)| \le \frac{9h^4}{384}\|f^{(4)}\|_C \le \frac{9}{384}\Big(\frac{3}{N}\Big)^4 \le \varepsilon = 10^{-6}$$

получаем $N \ge 38$. При $N = 5$ имеем $E < 0,0031$.

2.12. Свойства функции B_3 проверяются непосредственно.

2.13. $S(0,5) = 5/8$.

2.14. Достаточно проверить условия стыковки соседних многочленов:

$$p_{i-1}^{(r)}(x_i - 0) = p_i^{(r)}(x_i + 0), \quad r = 0, 1, 2.$$

2.15. Нет, нельзя.

2.16. Да, совпадает.

2.17. Функция S дважды непрерывно дифференцируема и на каждом подотрезке $[x_i, x_{i+1}]$ является кубическим многочленом, т. е. она является кубическим сплайном. Для доказательства обратного достаточно показать, что функции x^α, $\alpha = 0, 1, 2, 3$, $(x - x_i)_+^3$, $i = 1, 2, \ldots, N-1$ линейно независимы и образуют базис в пространстве кубических сплайнов из C^2 с узлами x_i, $i = 1, 2, \ldots, N-1$.

2.18. См. решение задачи 2.2.

2.19. Так как разделенные разности четвертого порядка равны нулю, то из задачи 2.18 следует, что $P_{i,3}(x) \equiv P_{i+1,3}(x)$, $i = 0, 1, \ldots, N-4$. Поэтому $P_{i,3}(x_j) = f_j$ для всех j.

2.20. Параметрический кубический сплайн $S(t) = (S_x(t), S_y(t))$ инвариантен относительно преобразований сдвига и растяжения/сжатия. Полагая $\tilde{t} = t\sqrt{3}/9$, исходные данные можно представить в виде

i	0	1	2	3
$\tilde{t}_i$	0	$1/3$	$2/3$	1
x_i	1	$-1/2$	$-1/2$	1
y_i	0	$\sqrt{3}/2$	$-\sqrt{3}/2$	0

Так как точки P_i являются равноотстоящими, то равномерная параметризация совпадает с параметризацией по суммарной длине хорд.

С учетом условий периодичности $M_{i+3} = M_i$, $f_{i+3} = f_i$, $i = 0, 1$ условия стыковки первой производной сплайна дают линейную систему

$$\mathbf{A}\tilde{\mathbf{M}} = \begin{pmatrix} 4 & 1 & 1 \\ 1 & 4 & 1 \\ 1 & 1 & 4 \end{pmatrix} \begin{pmatrix} \tilde{M}_1 \\ \tilde{M}_2 \\ \tilde{M}_3 \end{pmatrix} = \begin{pmatrix} -2f_1 + f_2 + f_3 \\ f_1 - 2f_2 + f_3 \\ f_1 + f_2 - 2f_3 \end{pmatrix} = \mathbf{f}, \quad \tilde{\mathbf{M}} = \frac{h^2}{6}\mathbf{M}.$$

Записывая эту систему отдельно для данных по x и y, получаем

$$\mathbf{A}\tilde{\mathbf{M}}_x = \mathbf{f}_x \quad \text{и} \quad \mathbf{A}\tilde{\mathbf{M}}_y = \mathbf{f}_y,$$

где $\mathbf{f}_x = \frac{3}{2}(1, 1, -1)^T$ и $\mathbf{f}_y = \frac{3\sqrt{3}}{2}(-1, 1, 0)^T$. Решая эти системы, последовательно находим $\tilde{\mathbf{M}}_x = \frac{1}{12}(5, 5, -7)^T$ и $\tilde{\mathbf{M}}_y = \frac{\sqrt{3}}{24}(-11, 7, 1)^T$.

Привлекая теперь формулы для составляющих сплайн кубических многочленов, для $\tilde{t} \in [\tilde{t}_i, \tilde{t}_{i+1}]$ получаем

$$\begin{aligned} S_x(\tilde{t}) &= x_i(1-u) + x_{i+1}u - u(1-u)\big[(2-u)\tilde{M}_{x,i} + (1+u)\tilde{M}_{x,i+1}\big], \\ S_y(\tilde{t}) &= y_i(1-u) + y_{i+1}u - u(1-u)\big[(2-u)\tilde{M}_{y,i} + (1+u)\tilde{M}_{y,i+1}\big], \end{aligned}$$

где $u = (\tilde{t} - \tilde{t}_i)/h$. Это позволяет выписать параметрический кубический сплайн в явном виде. В частности, находим $S_x(0,5) = -13/16$.

2.21. Функция $B_{j,3}$ непрерывно дифференцируема. Явная формула для квадратического B-сплайна имеет вид:

$$B_{j,2}(x) = \begin{cases} \dfrac{x - x_j}{x_{j+2} - x_j}\dfrac{x - x_j}{x_{j+1} - x_j} & \text{при} \quad x_j \le x \le x_{j+1}, \\ \dfrac{x - x_j}{x_{j+2} - x_j}\dfrac{x_{j+2} - x}{x_{j+2} - x_{j+1}} + \dfrac{x_{j+3} - x}{x_{j+3} - x_{j+1}}\dfrac{x - x_{j+1}}{x_{j+2} - x_{j+1}} & \\ & \text{при} \quad x_{j+1} \le x \le x_{j+2}, \\ \dfrac{x_{j+3} - x}{x_{j+3} - x_{j+1}}\dfrac{x_{j+3} - x}{x_{j+3} - x_{j+2}} & \text{при} \quad x_{j+2} \le x \le x_{j+3}, \\ 0 & \text{при} \quad x \le x_j \text{ или } x \ge x_{j+3}. \end{cases}$$

Глава 3

3.1. $I(y) = \sum\limits_{i=0}^{N}(a - y_i)^2$, $\dfrac{dI}{da} = 2\sum\limits_{i=0}^{N}(a - y_i) = 0$, $a = \dfrac{1}{N+1}\sum\limits_{i=0}^{N} y_i$.

3.2. $y(x) = a + bx, \qquad a = \dfrac{x_1 y_0 - x_0 y_1}{x_1 - x_0}, \qquad b = \dfrac{y_1 - y_0}{x_1 - x_0}$.

3.3. $a \approx 2,5929, \quad b \approx -0,32583, \quad c \approx 0,022738$.

3.4. а) $y = 48,49 - 2,9275x$; б) 19 машин.

3.5. а) $\eta = a + b\xi, \xi = x^{-1}, \eta = y^{-1}$; б) $\eta = a + b\xi + c\xi^2, \xi = x^{-1}, \eta = y$.

3.6. Для получения монотонно возрастающей ломаной узлы сгущают в области «большого градиента». Можно положить, например, $x_0 = 0$, $x_1 = 5$, $x_2 = 10$, $x_3 = 13$, $x_4 = 15$.

3.8. $x = -1, \quad y = \dfrac{20}{13}$.

Глава 4

4.1. $f''(\frac{1}{2}) = 0$, $f''(x_i) = \dfrac{1}{h^2}[f(x_{i-1}) - 2f(x_i) + f(x_{i+1})] - \dfrac{h^2}{12}f^{(4)}(\xi)$, $x_{i-1} \le \xi \le x_{i+1}$, $\dfrac{h^2}{12}\left|f^{(4)}(\xi)\right| \le \dfrac{\pi^4\sqrt{2}}{384} \approx 0,3587$.

$$
\begin{aligned}
4.2. \quad s'(t_i) &= \lambda_i \frac{s(t_i) - s(t_{i-1})}{h_{i-1}} + \mu_i \frac{s(t_{i+1}) - s(t_i)}{h_i}, \quad i = 1, 2, \ldots, N-1, \\
s'(t_0) &= (1 + \mu_1)\frac{s(t_1) - s(t_0)}{h_0} - \mu_1 \frac{s(t_2) - s(t_1)}{h_1}, \\
s'(t_N) &= -\lambda_{N-1}\frac{s(t_{N-1}) - s(t_{N-2})}{h_{N-2}} + (1 + \lambda_{N-1})\frac{s(t_N) - s(t_{N-1})}{h_{N-1}}, \\
&\mu_i = h_{i-1}(h_{i-1} + h_i)^{-1}, \quad \lambda_i = 1 - \mu_i, \quad h_i = t_{i+1} - t_i;
\end{aligned}
$$

Время в секундах	0	3	5	8	10	13
Скорость	79	82,4	74,2	76,8	69,4	71,2

4.3. $f'''(x_i) = \dfrac{1}{h^4}\big[f(x_i) - 3f(x_i + h) + 3f(x_i + 2h) - f(x_i + 3h)\big] = f'''(x_i + 3h)$.

4.4. $h = \sqrt[3]{24\varepsilon/M}$, $M = \max\limits_{x_{i-1} \le x \le x_{i+1}} |f^{(4)}(x)|$, $\varepsilon = \max(|\varepsilon_{i-1}|, |\varepsilon_i|, |\varepsilon_{i+1}|)$.

4.5. а) Составное правило прямоугольников требует $h < 0,03098$ и $N \ge 64$. Приближенное значение 0,405460.

б) Составное правило трапеций требует $h < 0,04382$ и $N \ge 46$. Приближенное значение 0,405471.

в) Составное правило Симпсона требует $h \le 0,44267$ и $N \ge 6$. Приближенное значение 0,405466.

4.6. В случае правила трапеций получаем следующие приближения: а) 0,265625; б) −0,2678571; в) 0,2280741; г) 0,1839397; д) −0,8666667; е) −0,6166667; ж) 0,2180895; з) 4,1432597.

Правило Симпсона дает: а) 0,1940104; б) −0,2670635; в) 0,1922453; г) 0,16240168; д) −0,7391053; е) −0,5518759; ж) 0,1513826; з) 2,5836964.

4.7. $|R_{3,i}(f)| \le \dfrac{h^5}{6480} \max\limits_{x_i \le x \le x_{i+1}} |f^{(4)}(x)|$.

4.8. Данная квадратурная формула является квадратурной формулой Гаусса, точной на кубических многочленах.

4.9. $c_0 = \frac{1}{4}$, $c_1 = \frac{3}{4}$, $x_1 = \frac{2}{3}$. Формула точна на многочленах второй степени.

4.10. $c_1 = \frac{1}{2}$, $x_0 = \frac{1}{2} - \frac{\sqrt{3}}{6}$, $x_1 = \frac{1}{2} + \frac{\sqrt{3}}{6}$. Формула точна на кубических многочленах.

4.11. Данная формула является квадратурной формулой Гаусса-Чебышева с узлами $x_i = \cos(2i-1)\pi/6$, $i = 1, 2, 3$ и коэффициентами $c_i = \pi/3$. Ее остаточный член имеет вид $R_3(f) = \dfrac{\pi}{6!2^5} f^{(6)}(\xi)$, $\xi \in [-1, 1]$. Поэтому формула точна на многочленах пятой степени.

4.12. $\dfrac{4}{3}[2f(-1) - f(0) + 2f(1)]$.

4.13. 3,28968.

4.14. Адаптивные квадратурные формулы дают:

$$\int_{0,1}^{2} \sin\frac{1}{x}\,dx = 1,1454 \quad \text{и} \quad \int_{0,1}^{2} \cos\frac{1}{x}\,dx = 0,67378.$$

Глава 5

5.1. Перестановка строк (столбцов) квадратной матрицы A достигается умножением ее на матрицу перестановок слева (справа). Норма последней равна 1.

5.2. Обозначим $\|\mathbf{x}\|_A = (\mathbf{A}\mathbf{x}, \mathbf{x})$. Покажем выполнение аксиом:

1. $\|\mathbf{x}\|_A \geq 0$ и $\|\mathbf{x}\|_A = 0$ тогда и только тогда, когда $\mathbf{x} = \mathbf{0}$;
2. $\|\lambda\mathbf{x}\|_A = |\lambda|\,\|\mathbf{x}\|_A$;
3. $\|\mathbf{x} + \mathbf{y}\|_A \leq \|\mathbf{x}\|_A + \|\mathbf{y}\|_A$.

Первая аксиома выполняется в силу условия $\mathbf{A} > 0$. Вторая аксиома следует из линейности скалярного произведения. Докажем третью аксиому.

Поскольку $\mathbf{A} = \mathbf{A}^T > 0$, то матрица $\mathbf{A}$ имеет систему собственных векторов $\mathbf{u}_1, \mathbf{u}_2, \ldots, \mathbf{u}_n$, образующих ортонормированный базис в $\mathbb{R}^n$ и отвечающих собственным значениям $0 < \lambda_1 < \lambda_2 < \ldots < \lambda_n$. Тогда, пользуясь неравенством Буняковского, имеем

$$\begin{aligned}(\mathbf{A}\mathbf{x}, \mathbf{y}) &= \left(\mathbf{A}\sum_{i=1}^{n}\alpha_i\mathbf{u}_i, \sum_{i=1}^{n}\beta_i\mathbf{u}_i\right) = \sum_{i=1}^{n}\lambda_i\alpha_i\beta_i \leq \left(\sum_{i=1}^{n}\lambda_i\alpha_i^2\right)^{1/2}\left(\sum_{i=1}^{n}\lambda_i\beta_i^2\right)^{1/2} = \\ &= \left(\sum_{i=1}^{n}\lambda_i(\alpha_i\mathbf{u}_i, \alpha_i\mathbf{u}_i)\right)^{1/2}\left(\sum_{i=1}^{n}\lambda_i(\beta_i\mathbf{u}_i, \beta_i\mathbf{u}_i)\right)^{1/2} = (\mathbf{A}\mathbf{x}, \mathbf{x})^{1/2}(\mathbf{A}\mathbf{y}, \mathbf{y})^{1/2},\end{aligned}$$

и поэтому

$$\begin{aligned}(\mathbf{A}(\mathbf{x} + \mathbf{y}), \mathbf{x} + \mathbf{y}) &= (\mathbf{A}\mathbf{x}, \mathbf{x}) + (\mathbf{A}\mathbf{x}, \mathbf{y}) + (\mathbf{A}\mathbf{y}, \mathbf{x}) + (\mathbf{A}\mathbf{y}, \mathbf{y}) \leq \\ &\leq (\mathbf{A}\mathbf{x}, \mathbf{x}) + 2(\mathbf{A}\mathbf{x}, \mathbf{x})^{1/2}(\mathbf{A}\mathbf{y}, \mathbf{y})^{1/2} + (\mathbf{A}\mathbf{y}, \mathbf{y}) = \left((\mathbf{A}\mathbf{x}, \mathbf{x})^{1/2} + (\mathbf{A}\mathbf{y}, \mathbf{y})^{1/2}\right)^2.\end{aligned}$$

Отсюда следует выполнение третьей аксиомы.

5.3. Пусть λ – собственное значение матрицы $\mathbf{A}$. Если λ – вещественное число, то ему соответствует вещественный собственный вектор $\mathbf{v}$. Тогда

$$\|\mathbf{A}\| = \sup_{\mathbf{u}\in\mathbf{R}^n,\ \mathbf{u}\neq\mathbf{0}} \frac{\|\mathbf{A}\mathbf{u}\|}{\|\mathbf{u}\|} \geq \frac{\|\mathbf{A}\mathbf{v}\|}{\|\mathbf{v}\|} = \frac{\|\lambda\mathbf{v}\|}{\|\mathbf{v}\|} = |\lambda|, \quad \text{т. е.} \quad |\lambda| \leq \|\mathbf{A}\|.$$

Пусть теперь имеем комплексное собственное значение $\lambda = \alpha + i\beta$ и ему соответствует комплексный собственный вектор $\mathbf{u} = \mathbf{w} + i\mathbf{v}$, где $\mathbf{w}$ и $\mathbf{v}$ – вещественные вектора. Тогда

$$\mathbf{A}\mathbf{u} = \mathbf{A}\mathbf{w} + i\mathbf{A}\mathbf{v} = (\alpha + i\beta)(\mathbf{w} + i\mathbf{v}) = \alpha\mathbf{w} - \beta\mathbf{v} + i(\beta\mathbf{w} + \alpha\mathbf{v}),$$

т. е. $\mathbf{A}\mathbf{w} = \alpha\mathbf{w} - \beta\mathbf{v}$, $\mathbf{A}\mathbf{v} = \beta\mathbf{w} + \alpha\mathbf{v}$. Домножая первое из этих равенств на α а второе на β и складывая их, получаем $\mathbf{A}(\alpha\mathbf{w} + \beta\mathbf{v}) = (\alpha^2 + \beta^2)\mathbf{w}$. Аналогично находим $\mathbf{A}(\alpha\mathbf{v} - \beta\mathbf{w}) = (\alpha^2 + \beta^2)\mathbf{v}$.

Обозначим $\rho^2 = \alpha^2 + \beta^2$. Тогда

$$\begin{aligned}\rho^2\|\mathbf{w}\| &= \|\mathbf{A}(\alpha\mathbf{w} + \beta\mathbf{v})\| \leq \|\mathbf{A}\|\,\|\alpha\mathbf{w} + \beta\mathbf{v}\| \leq \\ &\leq \|\mathbf{A}\|(|\alpha|\,\|\mathbf{w}\| + |\beta|\,\|\mathbf{v}\|) \leq \rho\|\mathbf{A}\|(\|\mathbf{w}\| + \|\mathbf{v}\|).\end{aligned}$$

Аналогично получаем $\rho^2\|\mathbf{v}\| \leq \rho\|\mathbf{A}\|(\|\mathbf{w}\| + \|\mathbf{v}\|)$. Тогда

$$\rho^2(\|\mathbf{w}\| + \|\mathbf{v}\|) \leq 2\rho\|\mathbf{A}\|(\|\mathbf{w}\| + \|\mathbf{v}\|)$$

и, следовательно, $\rho \leq 2\|\mathbf{A}\|$, где $\rho = |\lambda| = \sqrt{\alpha^2 + \beta^2}$. Поскольку λ^k – собственное значение матрицы $\mathbf{A}^k$, $k \geq 1$, то $\|\mathbf{A}^k\| \geq |\lambda|^k/2$ и $\|\mathbf{A}^k\|^{1/k} \geq |\lambda|/2^{1/k}$. По свойству нормы $\|\mathbf{A}^k\| \leq \|\mathbf{A}\|^k$ и, следовательно, $\|\mathbf{A}\| \geq |\lambda|/2^{1/k}$. Переходя к пределу по $k \to \infty$, имеем $\|\mathbf{A}\| \geq |\lambda|$.

5.4. $\max_i x_i^2 \leq \sum_{i=1}^n x_i^2 \leq \left(\sum_{i=1}^n |x_i|\right)^2$.

5.5. $\|\mathbf{Q}\|_2 = \sup\limits_{\mathbf{x}\neq\mathbf{0}} \dfrac{\|\mathbf{Q}\mathbf{x}\|_2}{\|\mathbf{x}\|_2} = \sup\limits_{\mathbf{x}\neq\mathbf{0}} \dfrac{(\mathbf{x}^T\mathbf{Q}^T\mathbf{Q}\mathbf{x})^{1/2}}{\|\mathbf{x}\|} = 1$. Аналогично имеем $\|\mathbf{Q}^T\|_2 = 1$. Поэтому $\mathrm{cond}_2(\mathbf{Q}) = \|\mathbf{Q}\|_2\,\|\mathbf{Q}^T\|_2 = 1$.

5.6. Если предположить, что матрица вырождена, т. е. $\det(\mathbf{A}) = 0$, то однородная система уравнений $\mathbf{A}\mathbf{x} = \mathbf{0}$ имеет нетривиальное решение $\mathbf{x} = (x_1, x_2, \ldots, x_n)^T$ и $\sum_j a_{ij}x_j = 0$, $i = 1, 2, \ldots, n$. Пусть $|x_k| \geq |x_i|$ для всех i. Тогда из k-го уравнения следует

$$|a_{kk}|\,|x_k| \leq \sum_{j\neq k} |a_{kj}|\,|x_j| \leq |x_k| \sum_{j\neq k} |a_{kj}|,$$

т. е. $|a_{kk}| \leq \sum_{j\neq k} |a_{kj}|$, что противоречит предположению.

5.7. Следуя [2], будем искать интересующую нас матрицу в виде

$$\mathbf{A} = \begin{pmatrix} a & b \\ c & d \end{pmatrix}, \quad \mathrm{cond}_2(\mathbf{A}) = \sqrt{\frac{\max_{1\le i\le n}\sigma_i}{\min_{1\le i\le n}\sigma_i}},$$

где σ_i – собственные значения матрицы

$$\mathbf{B} = \mathbf{A}^T\mathbf{A} = \begin{pmatrix} a^2+c^2 & ab+cd \\ ab+cd & b^2+d^2 \end{pmatrix}.$$

Пусть $\mathrm{tr}(\mathbf{B})$ – след матрицы $\mathbf{B}$, т. е. сумма ее диагональных элементов, а $\det(\mathbf{B})$ – определитель матрицы $\mathbf{B}$. Характеристическое уравнение для матрицы $\mathbf{B}$

$$\sigma^2 - \mathrm{tr}(\mathbf{B})\sigma + \det(\mathbf{B}) = 0$$

имеет корни

$$\sigma_{1,2} = \frac{1}{2}\Big[\mathrm{tr}(\mathbf{B}) \pm \sqrt{\mathrm{tr}^2(\mathbf{B}) - 4\det(\mathbf{B})}\,\Big].$$

Так как $\mathrm{tr}(\mathbf{B}) > 0$, то

$$\begin{aligned}\mathrm{cond}_2(\mathbf{A}) &= \sqrt{\frac{\mathrm{tr}(\mathbf{B}) + \sqrt{\mathrm{tr}^2(\mathbf{B}) - 4\det(\mathbf{B})}}{\mathrm{tr}(\mathbf{B}) - \sqrt{\mathrm{tr}^2(\mathbf{B}) - 4\det(\mathbf{B})}}} = \frac{\mathrm{tr}(\mathbf{B}) + \sqrt{\mathrm{tr}^2(\mathbf{B}) - 4\det(\mathbf{B})}}{\sqrt{4\det(\mathbf{B})}} = \\ &= \frac{\mathrm{tr}(\mathbf{B})}{2\sqrt{\det(\mathbf{B})}} + \sqrt{\frac{\mathrm{tr}^2(\mathbf{B})}{4\det(\mathbf{B})} - 1}.\end{aligned}$$

Таким образом, $\mathrm{cond}_2(\mathbf{A}) \to \max$, если $\mathrm{tr}^2(\mathbf{B})/\det(\mathbf{B}) \to \max$. Поскольку

$$\begin{aligned}\mathrm{tr}^2(\mathbf{B}) &= (a^2+b^2+c^2+d^2)^2, \quad \det(\mathbf{B}) = (a^2+c^2)(b^2+d^2) - (ab+cd)^2 = \\ &= a^2d^2 + b^2c^2 - 2abcd = (ad-bc)^2 = \begin{vmatrix} a & b \\ c & d \end{vmatrix}^2,\end{aligned}$$

то нам надо максимизировать величину

$$(a^2+b^2+c^2+d^2)\begin{vmatrix} a & b \\ c & d \end{vmatrix}^{-1}.$$

Итак, решаем задачу

$$\begin{vmatrix} a & b \\ c & d \end{vmatrix} = \pm 1, \quad a^2+b^2+c^2+d^2 \to \max.$$

В данном случае можно воспользоваться любой из матриц

$$\mathbf{A}_1 = \begin{pmatrix} n+2 & n+1 \\ n+1 & n \end{pmatrix}, \quad \mathbf{A}_2 = \begin{pmatrix} n+1 & n+2 \\ n & n+1 \end{pmatrix},$$

$$\mathbf{A}_3 = \begin{pmatrix} n+1 & n \\ n+2 & n+1 \end{pmatrix}, \quad \mathbf{A}_4 = \begin{pmatrix} n & n+1 \\ n+1 & n+2 \end{pmatrix}.$$

5.8. $\mathrm{cond}_2(\mathbf{A}+\tau\mathbf{E}) = (\lambda_n+\tau)/(\lambda_1+\tau) = 1 + (\lambda_n-\lambda_1)/(\lambda_1+\tau)$.

5.9. Матрица системы является плохообусловленной с $\mathrm{cond}(\mathbf{A}) \approx 198^2$. Поэтому из малости невязки не следует малость ошибки. Так как точное решение $x^* = (1;1)^T$, то лучший результат дает компьютер AMD64.

5.10. Значения детерминанта $\det(\mathbf{A})$, получаемые по формулам (1.12) и (1.15), отличаются знаком, что вызвано перестановкой строк при гауссовом исключении с выбором ведущего элемента по столбцу.

5.11. а) Коэффициенты в первой строке «стреловидной» матрицы зануляем, используя строки с последней по вторую; в результате матрица приводится к нижней треугольной форме;

б) используя строки со второй по предпоследнюю, исключаем «промежуточные» коэффициенты в первой и последней строках; уничтожая затем коэффициенты последнего столбца, приводим матрицу к нижней треугольной форме;

в) начиная с первой строки, зануляем элементы поддиагонали; возвращаясь, исключаем элементы первого столбца; получаем матрицу в верхней треугольной форме.

5.12. а) По определению матрицей отражения называется матрица следующего вида $\mathbf{H} = \mathbf{E} - 2\mathbf{w}\mathbf{w}^T/\|\mathbf{w}\|_2^2$; возьмем вектор $\mathbf{w} = (\bar{c}, \bar{s})$, $\bar{c} = \cos(-\varphi/2)$, $\bar{s} = \sin(-\varphi/2)$; тогда $\|\mathbf{w}\|_2^2 = 1$ и

$$\mathbf{H} = \mathbf{E} - 2\mathbf{w}\mathbf{w}^T = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix} - 2\begin{pmatrix} \bar{c}^2 & -\bar{s}\,\bar{c} \\ -\bar{s}\,\bar{c} & \bar{s}^2 \end{pmatrix} = \begin{pmatrix} \bar{s}^2-\bar{c}^2 & 2\bar{s}\,\bar{c} \\ 2\bar{s}\,\bar{c} & \bar{c}^2-\bar{s}^2 \end{pmatrix} = \begin{pmatrix} -c & s \\ s & c \end{pmatrix},$$

где $s = \sin\varphi$, $c = \cos\varphi$;

б) Умножая на матрицу отражения, получаем нуль в позиции (2,1)

$$\mathbf{H}\begin{pmatrix} 1 \\ 1 \end{pmatrix} = \begin{pmatrix} -c & s \\ s & c \end{pmatrix}\begin{pmatrix} 1 \\ 1 \end{pmatrix} = \begin{pmatrix} * \\ 0 \end{pmatrix},$$

т. е. $s + c = 0$ или $c = -s = 1/\sqrt{2}$. Тогда

$$\begin{aligned} \mathbf{H}\mathbf{A} &= -\frac{1}{\sqrt{2}}\begin{pmatrix} 1 & 1 \\ 1 & -1 \end{pmatrix}\begin{pmatrix} 1 & 2 \\ 1 & 3 \end{pmatrix} = -\frac{1}{\sqrt{2}}\begin{pmatrix} 2 & 5 \\ 0 & -1 \end{pmatrix}, \\ \mathbf{H}\mathbf{b} &= -\frac{1}{\sqrt{2}}\begin{pmatrix} 1 & 1 \\ 1 & -1 \end{pmatrix}\begin{pmatrix} 1 \\ 2 \end{pmatrix} = -\frac{1}{\sqrt{2}}\begin{pmatrix} 3 \\ -1 \end{pmatrix}. \end{aligned}$$

В результате приходим к системе $\begin{pmatrix} 2 & 5 \\ 0 & -1 \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \end{pmatrix} = \begin{pmatrix} 3 \\ -1 \end{pmatrix}$, имеющей решение $x_2 = 1$ и $x_1 = -1$.

5.13. Так как

$$\bar{\mathbf{e}}_2 = \mathbf{H}\mathbf{e}_2 = \begin{pmatrix} -c & s \\ s & c \end{pmatrix} \begin{pmatrix} 0 \\ 1 \end{pmatrix} = \begin{pmatrix} s \\ c \end{pmatrix}, \ \bar{\mathbf{e}}_1 = \mathbf{H}\mathbf{e}_1 = \begin{pmatrix} -c & s \\ s & c \end{pmatrix} \begin{pmatrix} 1 \\ 0 \end{pmatrix} = \begin{pmatrix} -c \\ s \end{pmatrix},$$

то векторы $\mathbf{e}_1$ и $\mathbf{e}_2$ зеркально отражаются относительно прямой, проходящей через начало координат и образующей угол $\varphi/2$ с вектором $\mathbf{e}_2$, и переходят в ортогональные вектора $\bar{\mathbf{e}}_1$ и $\bar{\mathbf{e}}_2$. Так как для произвольного вектора $\mathbf{x} = (x_1; x_2)^T$ имеем $\bar{\mathbf{x}} = \mathbf{H}\mathbf{x} = \mathbf{H}(x_1\mathbf{e}_1 + x_2\mathbf{e}_2) = x_1\bar{\mathbf{e}}_1 + x_2\bar{\mathbf{e}}_2$, то вектор $\bar{\mathbf{x}}$ является зеркальным отражением вектора $\mathbf{x}$ относительно той же прямой.

Поскольку

$$\bar{\mathbf{e}}_1 = \mathbf{J}\mathbf{e}_1 = \begin{pmatrix} c & s \\ -s & c \end{pmatrix} \begin{pmatrix} 1 \\ 0 \end{pmatrix} = \begin{pmatrix} c \\ -s \end{pmatrix}, \ \bar{\mathbf{e}}_2 = \mathbf{J}\mathbf{e}_2 = \begin{pmatrix} c & s \\ -s & c \end{pmatrix} \begin{pmatrix} 0 \\ 1 \end{pmatrix} = \begin{pmatrix} s \\ c \end{pmatrix},$$

то векторы $\mathbf{e}_1$ и $\mathbf{e}_2$ поворачиваются на угол φ по часовой стрелке. Поэтому вектор $\bar{\mathbf{x}} = \mathbf{J}\mathbf{x}$ является результатом поворота вектора $\mathbf{x}$ на угол φ.

5.14. Для матриц перехода в методах Якоби и Зейделя имеем:

$$\mathbf{C}_{\text{Я}} = -\mathbf{D}^{-1}(\mathbf{L}+\mathbf{U}) = \begin{pmatrix} 0 & -\dfrac{a_{12}}{a_{11}} \\ -\dfrac{a_{21}}{a_{22}} & 0 \end{pmatrix}, \quad \lambda^{\text{Я}}_{1,2} = \pm\sqrt{\frac{a_{12}a_{21}}{a_{11}a_{22}}};$$

$$\mathbf{C}_{\text{З}} = -(\mathbf{D}+\mathbf{L})^{-1}\mathbf{U} = \begin{pmatrix} 0 & -\dfrac{a_{12}}{a_{11}} \\ 0 & \dfrac{a_{12}a_{21}}{a_{11}a_{22}} \end{pmatrix}, \quad \lambda^{\text{З}}_1 = 0, \quad \lambda^{\text{З}}_2 = \frac{a_{12}a_{21}}{a_{11}a_{22}}.$$

5.15. Рассмотрим матрицу $\mathbf{A} = \begin{pmatrix} 2 & 4 & -4 \\ 3 & 3 & 3 \\ 10 & 10 & 5 \end{pmatrix}$. Тогда для матриц перехода в методах Якоби и Зейделя соответственно имеем:

$$\mathbf{C}_{\text{Я}} = -\begin{pmatrix} 2 & 0 & 0 \\ 0 & 3 & 0 \\ 0 & 0 & 5 \end{pmatrix}^{-1} \begin{pmatrix} 0 & 4 & -4 \\ 3 & 0 & 3 \\ 10 & 10 & 0 \end{pmatrix} = \begin{pmatrix} 0 & -2 & 2 \\ -1 & 0 & -1 \\ -2 & -2 & 0 \end{pmatrix},$$

$$\mathbf{C}_{\text{З}} = -\begin{pmatrix} 2 & 0 & 0 \\ 3 & 3 & 0 \\ 10 & 10 & 5 \end{pmatrix}^{-1} \begin{pmatrix} 0 & 4 & -4 \\ 0 & 0 & 3 \\ 0 & 0 & 0 \end{pmatrix} = \begin{pmatrix} 0 & 2 & -2 \\ 0 & -2 & 3 \\ 0 & 0 & -2 \end{pmatrix}.$$

Так как $\det(\mathbf{C}_{\text{Я}} - \lambda\mathbf{E}) = -\lambda^3$, $\det(\mathbf{C}_{\text{З}} - \lambda\mathbf{E}) = -\lambda(2+\lambda)^2$, то метод Якоби сходится, а метод Зейделя расходится.

5.16. Рассмотрим итерационный процесс

$$\mathbf{x}^{(k+1)} = \mathbf{x}^{(k)} - \tau_k(\mathbf{A}\mathbf{x}^{(k)} - \mathbf{b}), \quad k = 0, 1, \dots .$$

Пусть $\mathbf{x}^*$ – точное решение системы $\mathbf{A}\mathbf{x} = \mathbf{b}$. Для погрешности $\mathbf{w}^{(k)} = \mathbf{x}^{(k)} - \mathbf{x}^*$ получаем

$$\mathbf{w}^{(k+1)} = (\mathbf{E} - \tau_k\mathbf{A})\mathbf{w}^{(k)} = (\mathbf{E} - \tau_k\mathbf{A}) \dots (\mathbf{E} - \tau_0\mathbf{A})\mathbf{w}^{(0)}.$$

Собственные векторы $\mathbf{u}_1, \mathbf{u}_2, \dots, \mathbf{u}_n$ симметрической матрицы $\mathbf{A}$, соответствующие собственным значениям $\lambda_1, \lambda_2, \dots, \lambda_n$, образуют базис в $\mathbb{R}^n$. При $\mathbf{w}^{(0)} = \sum_{i=1}^n c_i^{(0)}\mathbf{u}_i$ и $\tau_k = 1/\lambda_{k+1}$, $k = 0, 1, \dots, n-1$, находим

$$\mathbf{w}^{(k+1)} = \sum_{i=1}^{n}(1 - \tau_k\lambda_i) \dots (1 - \tau_0\lambda_i)c_i^{(0)}\mathbf{u}_i = \sum_{i=k+2}^{n}\Big(1 - \frac{\lambda_i}{\lambda_{k+1}}\Big) \dots \Big(1 - \frac{\lambda_i}{\lambda_1}\Big)c_i^{(0)}\mathbf{u}_i.$$

Таким образом, $\mathbf{w}^{(n)} = \mathbf{0}$, т. е. через n шагов будет получено точное решение системы $\mathbf{A}\mathbf{x} = \mathbf{b}$.

5.17. Запишем уравнение для погрешности в методе релаксации

$$(\omega\mathbf{L} + \mathbf{D})\mathbf{w}^{(k+1)} = \big((1-\omega)\mathbf{D} - \omega\mathbf{U}\big)\mathbf{w}^{(k)}.$$

Умножим это равенство на матрицу $\mathbf{D}^{-1}$. Тогда матрица перехода принимает вид

$$\mathbf{C} = (\mathbf{E} + \omega\mathbf{D}^{-1}\mathbf{L})^{-1}\big((1-\omega)\mathbf{E} - \omega\mathbf{D}^{-1}\mathbf{U}\big).$$

Рассмотрим ее характеристический многочлен $d(\lambda) = \det(\mathbf{C} - \lambda\mathbf{E})$. По теореме Виета имеем $(-1)^n d(0) = \prod_{i=1}^n \lambda_i(\mathbf{C})$. Так как треугольные матрицы $\mathbf{D}^{-1}\mathbf{L}$ и $\mathbf{D}^{-1}\mathbf{U}$ имеют на диагонали нули, то

$$d(0) = \det(\mathbf{C}) = (1-\omega)^n.$$

Отсюда следует оценка сходимости

$$1 > \max_i |\lambda_i(\mathbf{C})| \geq \Big|\prod_{i=1}^{n}\lambda_i(\mathbf{C})\Big|^{1/n} = |\det(\mathbf{C})|^{1/n} = |1-\omega|.$$

5.18. Матрицы перехода в методах Якоби, Зейделя и релаксации:

$$\mathbf{C}_{\text{Я}} = -\mathbf{D}^{-1}(\mathbf{L} + \mathbf{U}) = \begin{pmatrix} 0 & -1/4 \\ -1/4 & 0 \end{pmatrix}, \quad \lambda_{1,2}^{\text{Я}} = \pm 1/4;$$

$$\mathbf{C}_{\text{З}} = -(\mathbf{D}+\mathbf{L})^{-1}\mathbf{U} = \begin{pmatrix} 0 & -1/4 \\ 0 & 1/16 \end{pmatrix}, \quad \lambda_1^{\text{З}} = 0, \quad \lambda_2^{\text{З}} = 1/16;$$

$$\mathbf{C}_{\text{Р}} = (\omega\mathbf{L}+\mathbf{D})^{-1}[(1-\omega)\mathbf{D} - \omega\mathbf{U}] = \begin{pmatrix} 1-\omega & -\omega/4 \\ -\omega(1-\omega)/4 & \omega^2/4+1-\omega \end{pmatrix},$$

$\omega_{\text{опт}} = 8(4-\sqrt{15}), \quad \lambda^{\text{Р}} = \omega_{\text{опт}} - 1 = 31 - 8\sqrt{15} \approx 0,0161 \approx 1/62.$

5.19. 1) $0 < \tau < 1/4,9$; 2) $0 < \tau < 2/9$; 3) $0 < \tau < 0,5$; 4) $0 < \tau < 0,4$.

5.20. Матрица перехода в методе Якоби $\mathbf{C} = \mathbf{D}^{-1}(\mathbf{L}+\mathbf{U})$ для матриц $\mathbf{A}_1$ и $\mathbf{A}_2$ имеет собственные значения: $\lambda_{1,2}(\mathbf{C_1}) = \pm 1/4$, $\lambda_{1,2}(\mathbf{C_2}) = \pm 1/2$, т. е. метод Якоби сходится быстрее для матрицы с меньшим диагональным преобладанием.

5.21. Перепишем метод относительно погрешности $\mathbf{w}^{(k)} = \mathbf{x}^{(k)} - \mathbf{x}^*$ в виде

$$(\mathbf{E}+\alpha\tau\mathbf{A})\mathbf{w}^{(k+1)} = \big(\mathbf{E} - (1-\alpha)\tau\mathbf{A}\big)\mathbf{w}^{(k)}.$$

Используя совпадение собственных векторов матриц слева и справа, выразим собственные значения матрицы перехода $\mathbf{C}$ через собственные значения матрицы $\mathbf{A}$:

$$\lambda(\mathbf{C}) = \frac{1-(1-\alpha)\tau\lambda}{1+\alpha\tau\lambda}.$$

Тогда из условия сходимости метода $|\lambda(\mathbf{C})| < 1$ следует неравенство

$$1 - \frac{2}{\tau\lambda} < 2\alpha \quad \text{или} \quad \alpha \geq 1/2.$$

Глава 6

6.1. в) вещественные собственные значения: 3, $[-2;2]$, $[-7;-5]$.

6.2. а) Матрица $\mathbf{A}$ подобна вещественной симметрической матрице: $\mathbf{A} = \mathbf{D}\mathbf{B}\mathbf{D}^{-1}$, где $\mathbf{D}$ – диагональная матрица с элементами

$$d_{11} = 1, \quad d_{ii} = \sqrt{\frac{a_2 a_3 \dots a_i}{c_1 c_1 \dots c_{i-1}}}, \quad i = 2,3,\dots,n,$$

а $\mathbf{B}$ – трехдиагональная вещественная матрица с элементами:

$$b_{ii} = b_i, \quad i = 1,2,\dots,n, \quad b_{i,i+1} = b_{i+1,i} = \sqrt{a_{i+1}c_i}, \quad i = 1,2,\dots,n-1.$$

б) Из теоремы Гершгорина следует, что для собственных значений матрицы $\mathbf{A}$ верно неравенство $|\lambda(\mathbf{A})| \leq 1$. Случай $|\lambda(\mathbf{A})| = 1$ рассмотрен в [12].

6.3. Симметрическая матрица $\mathbf{A}$ положительно определена, так как она имеет диагональное преобладание и ее диагональные элементы положительны (см. теорему 6.1).

6.4. Предположим противное. Пусть $\mathbf{Ax} \geq \mathbf{0}$, но среди компонент вектора $\mathbf{x}$ есть отрицательные и $|x_k| = \max_{1\leq i\leq n} |x_i|$. Возьмем вектор $\mathbf{y} = (|x_k|; |x_k|; \ldots; |x_k|)^T$. Так как

$$\sum_{j=1}^{n} a_{ij} y_j = |x_k| \sum_{j=1}^{n} a_{ij} > 0, \quad i = 1, 2, \ldots, n,$$

то $\mathbf{Ay} > \mathbf{0}$ и, следовательно, $\mathbf{A}(\mathbf{x} + \mathbf{y}) = \mathbf{Ax} + \mathbf{Ay} > \mathbf{0}$. С другой стороны

$$\sum_{j=1}^{n} a_{kj}(x_j + y_j) = \sum_{j=1}^{n} a_{kj}(x_j + |x_k|) \leq 0.$$

Полученное противоречие доказывает утверждение.

Пусть теперь $\mathbf{Ax} = \mathbf{e}_i$, $i = 1, 2, \ldots, n$. Тогда $\mathbf{x} = \mathbf{A}^{-1}\mathbf{e}_i \geq \mathbf{0}$, т. е. i-й столбец матрицы $\mathbf{A}^{-1}$ содержит только неотрицательные компоненты.

6.5. Собственные значения $\mathbf{A} + \Delta\mathbf{A}$: $\hat{\lambda} = [3 - \varepsilon \pm \sqrt{1 - 838\varepsilon + \varepsilon^2}]/2$. Здесь $\mathbf{V} = \begin{pmatrix} 9 & 10 \\ 10 & 11 \end{pmatrix}$, $\mathbf{V}^{-1} = \begin{pmatrix} -11 & 10 \\ 10 & -9 \end{pmatrix}$, $\|\mathbf{V}\|_1 = \|\mathbf{V}^{-1}\|_1 = 21$, $\mathrm{cond}_1(\mathbf{V}) = 441$, $\|\Delta\mathbf{A}\|_1 = \varepsilon$. При $\varepsilon = 0,001$ имеем $\hat{\lambda} \approx 1,701; 1,298$. Изменение на $0,001$ в двух элементах матрицы $\mathbf{A}$ дает изменение примерно на 0,3 в обоих собственных значениях. Согласно теореме Бауера-Файка получаем оценку $|\hat{\lambda} - \lambda| \leq \mathrm{cond}_1(\mathbf{V})\|\Delta\mathbf{A}\|_1 = 441\varepsilon = 0,441$.

6.6. Здесь $|\lambda_2/\lambda_3| = 5,1/5,2 \approx 0,981$ и ошибка убывает очень медленно. Матрица $\mathbf{B} = \mathbf{A} - 4\mathbf{E}$ имеет собственные значения $\hat{\lambda}_1 = -3$, $\hat{\lambda}_2 = 1,1$, $\hat{\lambda}_3 = -3$ и $|\hat{\lambda}_1/\hat{\lambda}_3| \approx 0,325$.

6.7. а) $\lambda = 9,701562$; $\mathbf{x}_1 = (-0,35078; 1,00000; -0,35078; 1,00000)^T$; б) $\lambda = 11$; $\mathbf{x}_1 = (0; -1; 0; 1)^T$.

6.8. 3,298431.

6.9. Находим $\mathbf{A}^5\mathbf{u}_0^{(1)} = \begin{pmatrix} 244 \\ -242 \end{pmatrix}$, $\mathbf{A}^5\mathbf{u}_0^{(2)} = \begin{pmatrix} 1 \\ 1 \end{pmatrix}$, $\mathbf{A}^5\mathbf{u}_0^{(3)} = \mathbf{A}^5\mathbf{u}_0^{(2)} + \dfrac{\varepsilon}{2}\mathbf{A}^5\mathbf{u}_0^{(1)} = \begin{pmatrix} 1 \\ 1 \end{pmatrix} + \dfrac{\varepsilon}{2}\begin{pmatrix} 244 \\ -242 \end{pmatrix}$. Матрица $\mathbf{A}$ имеет собственные значения $\lambda_1 = 1$, $\lambda_2 = 3$ и собственные векторы $\mathbf{x}_1 = (1, 1)^T/\sqrt{2}$, $\mathbf{x}_1 = (1, -1)^T/\sqrt{2}$. В первом случае с точностью до нормировки получен собственный вектор $\mathbf{x}_2$, отвечающий максимальному собственному значению. Рост величины $\|\mathbf{A}^5\mathbf{u}_0^{(1)}\|_2$ означает необходимость нормировки этого вектора. Во втором случае начальным приближением является вектор $\sqrt{2}\mathbf{x}_1$. Поэтому $\mathbf{A}^5\mathbf{u}_0^{(2)} = \sqrt{2}\mathbf{u}_0^{(2)}$. В третьем случае имеем «плохое» начальное приближение, что приводит к увеличению числа итераций. Полученное приближение уже «достаточно далеко» от собственного вектора $\mathbf{x}_1$, но еще «недостаточно близко» к собственному вектору $\mathbf{x}_2$.

6.10. Собственные векторы матрицы $\mathbf{A}$: $\mathbf{x}_1 = \frac{1}{\sqrt{2}}(1;1)$, $\mathbf{x}_2 = \frac{1}{\sqrt{2}}(1;-1)$. Получаем а) $\mathbf{A}^3\mathbf{u}_0 = (76;-49)^T$, итерации сходятся к собственному вектору $\mathbf{x}_2$, отвечающему собственному значению $\lambda = 5$. б) $(\mathbf{A}^{-1})^3\mathbf{u}_0 = 15^{-3}(76;49)^T$, $\mathbf{A}^{-1} = \frac{1}{15}\begin{pmatrix}4 & 1\\ 1 & 4\end{pmatrix}$, итерации сходятся к собственному вектору $\mathbf{x}_1$, отвечающему собственному значению $\lambda = 3$. В обоих случаях а) и б) нужна нормировка. в) $\alpha = 4$, $\mathbf{B} = \mathbf{B}^{-1} = \mathbf{A} - 4\mathbf{E} = \begin{pmatrix}0 & -1\\ -1 & 0\end{pmatrix}$, метод циклит.

6.11. Характеристический многочлен матрицы $(p-\lambda)^4 - 3(p-\lambda)^2 + 1 = 0$. Матрица $\mathbf{A}$ имеет простые собственные значения:

$$\lambda_{1,4} = p \pm \sqrt{\frac{3+\sqrt{5}}{2}}; \quad \lambda_{2,3} = p \pm \sqrt{\frac{3-\sqrt{5}}{2}}.$$

Собственные векторы матрицы $\mathbf{A}$:

$$\mathbf{e}_{1,4} = \frac{1}{\sqrt{5+\sqrt{5}}}\Big(1; \pm\frac{1+\sqrt{5}}{2}; \frac{1+\sqrt{5}}{2}; \pm 1\Big);$$
$$\mathbf{e}_{2,3} = \frac{1}{\sqrt{5+\sqrt{5}}}\Big(\frac{1+\sqrt{5}}{2}; \pm 1; -1; \mp\frac{1+\sqrt{5}}{2}\Big);$$
$$(\mathbf{e}_i, \mathbf{e}_j) = 0 \quad \text{при} \quad i \neq j; \quad \text{и} \quad (\mathbf{e}_i, \mathbf{e}_i) = 1.$$

6.12. Для приведения матрицы $\mathbf{A}$ к диагональному виду требуется выполнить только два шага метода вращений Якоби. Матрицы вращений:

$$\mathbf{R}_1 = \begin{pmatrix}\frac{1}{\sqrt{2}} & 0 & \frac{1}{\sqrt{2}}\\ 0 & 1 & 0\\ -\frac{1}{\sqrt{2}} & 0 & \frac{1}{\sqrt{2}}\end{pmatrix}, \quad \mathbf{R}_2 = \begin{pmatrix}1 & 0 & 0\\ 0 & \frac{2}{\sqrt{6}} & -\frac{1}{\sqrt{3}}\\ 0 & \frac{1}{\sqrt{3}} & \frac{2}{\sqrt{6}}\end{pmatrix}, \quad \lambda_1 = -2, \quad \lambda_2 = 6, \quad \lambda_3 = 3.$$

6.13. Матрица $\mathbf{A}$ имеет кратные собственные значения $\lambda_1 = \lambda_2 = \lambda_3 = p-1$; $\lambda_4 = p+3$. Собственные векторы матрицы $\mathbf{A}$ линейно независимы и образуют ортонормированный базис в $\mathbb{R}^4$:

$$\mathbf{e}_1 = \frac{1}{\sqrt{2}}\big(1;-1;0;0\big); \quad \mathbf{e}_2 = \frac{1}{\sqrt{6}}\big(-1;-1;2;0\big);$$
$$\mathbf{e}_3 = \frac{1}{\sqrt{12}}\big(-1;-1;-1;3\big); \quad \mathbf{e}_4 = \frac{1}{2}\big(1;1;1;1\big).$$

6.14. $\mathbf{R}_1 = \frac{1}{5}\begin{pmatrix}3 & 4 & 0\\ -4 & 3 & 0\\ 0 & 0 & 5\end{pmatrix}$, $\mathbf{R}_2 = \frac{1}{5}\begin{pmatrix}4 & 0 & 3\\ 0 & 5 & 0\\ -3 & 0 & 4\end{pmatrix}$, $\mathbf{R}_3 = \frac{1}{5}\begin{pmatrix}5 & 0 & 0\\ 0 & 3 & 4\\ 0 & -4 & 3\end{pmatrix}$,

$$\mathbf{Q} = \mathbf{R}_1^T\mathbf{R}_2^T\mathbf{R}_3^T = \frac{1}{125}\begin{pmatrix} 60 & -96 & 53 \\ 80 & -3 & -96 \\ 75 & 80 & 60 \end{pmatrix},\ \mathbf{R} = 125\begin{pmatrix} 3 & 2 & 3 \\ 0 & 1 & 1 \\ 0 & 0 & 2 \end{pmatrix}.$$

6.15. $\mathbf{Q}_1\mathbf{Q}_2\mathbf{Q}_2^T\mathbf{Q}_1^T = \mathbf{E}$, т. е. $(\mathbf{Q}_1\mathbf{Q}_2)^{-1} = \mathbf{Q}_2^T\mathbf{Q}_1^T = (\mathbf{Q}_1\mathbf{Q}_2)^T$.

6.16. а) $\mathbf{H}_1 = \frac{1}{15}\begin{pmatrix} -10 & -10 & -5 \\ -10 & 11 & -2 \\ -5 & -2 & 14 \end{pmatrix}$, $\mathbf{H}_2 = \frac{1}{5}\begin{pmatrix} 5 & 0 & 0 \\ 0 & -3 & -4 \\ 0 & -4 & 3 \end{pmatrix}$,

$$\mathbf{Q} = \mathbf{H}_1\mathbf{H}_2 = \frac{1}{3}\begin{pmatrix} -2 & 2 & 1 \\ -2 & -1 & -2 \\ -1 & -2 & 2 \end{pmatrix},\ \mathbf{R} = \mathbf{H}_2\mathbf{H}_1\mathbf{A} = \begin{pmatrix} -3 & 1 & -5 \\ 0 & -5 & -10 \\ 0 & 0 & 10 \end{pmatrix}.$$

б) $\mathbf{Q} = \frac{1}{3}\begin{pmatrix} -1 & 2 & 2 \\ -2 & 1 & -2 \\ -2 & -2 & 1 \end{pmatrix}$, $\mathbf{R} = 3\begin{pmatrix} -1 & 1 & -1 \\ 0 & 1 & -1 \\ 0 & 0 & 1 \end{pmatrix}$.

6.17. Характеристическое уравнение

$$\det(\mathbf{A} - \lambda\mathbf{E}) = \lambda^2 - 2(\beta+\delta)\lambda + \alpha\beta - \gamma^2 = 0$$

имеет корни $\lambda_{1,2} = \beta + \delta \pm \sqrt{\delta^2 + \gamma^2}$. Ближайшее к β собственное значение вычисляется по формуле $\lambda = \beta + \delta - \operatorname{sign}(\delta)\sqrt{\delta^2 + \gamma^2}$.

6.18. $\mathbf{T} = \begin{pmatrix} 1 & \sqrt{10} & 0 \\ \sqrt{10} & -0,4 & 3,2 \\ 0 & 3,2 & 4,4 \end{pmatrix}$, $\mathbf{Q} = \frac{1}{\sqrt{10}}\begin{pmatrix} 1 & 3 & 0 \\ \sqrt{10} & 0 & 0 \\ 0 & -1 & 3 \end{pmatrix}$.

6.19. $\lambda_{min} = \min\limits_{\mathbf{x}\neq\mathbf{0}} \dfrac{(\mathbf{Ax},\mathbf{x})}{(\mathbf{x},\mathbf{x})} \le \min\limits_{1\le i\le n} \dfrac{(\mathbf{Ae}_i,\mathbf{e}_i)}{(\mathbf{e}_i,\mathbf{e}_i)} = \min\limits_{1\le i\le n}(a_{ii})$,

$\lambda_{max} = \max\limits_{\mathbf{x}\neq\mathbf{0}} \dfrac{(\mathbf{Ax},\mathbf{x})}{(\mathbf{x},\mathbf{x})} \ge \max\limits_{1\le i\le n} \dfrac{(\mathbf{Ae}_i,\mathbf{e}_i)}{(\mathbf{e}_i,\mathbf{e}_i)} = \max\limits_{1\le i\le n}(a_{ii})$,

6.20. Симметрическая матрица $\mathbf{B}$ имеет полную систему линейно независимых собственных векторов $\{\mathbf{u}_i\}$. Для произвольного вектора $\mathbf{x} \in \mathbb{R}^n$ здесь имеем $\mathbf{ABx} = \mathbf{AB}(\sum_i c_i\mathbf{u}_i) = \mathbf{A}(\sum_i c_i\lambda_i\mathbf{u}_i) = \sum_i d_i\mathbf{Au}_i$, где $c_i\lambda_i = d_i$. Так как симметрическая матрица $\mathbf{A}$ положительно определена, то векторы $\mathbf{Au}_i$ будут линейно независимы и образуют полную систему собственных векторов матрицы $\mathbf{AB}$.

6.21. Симметрическая матрица $\mathbf{B} = \mathbf{QAQ}^{-1}$ имеет полную систему линейно независимых собственных векторов $\mathbf{u}_1, \mathbf{u}_2, \ldots, \mathbf{u}_n$, отвечающих собственным значениям λ_i, $i = 1, 2, \ldots, n$. Произвольный вектор $\mathbf{x}$ разложим по этой системе. Тогда $\mathbf{Bx} = \mathbf{QAQ}^{-1}\mathbf{x} = \mathbf{B}(\sum_{i=1}^n c_i\mathbf{u}_i) = \sum_{i=1}^n c_i\lambda_i\mathbf{u}_i$ и для вектора $\mathbf{y} = \mathbf{Q}^{-1}\mathbf{x}$ получаем $\mathbf{Ay} = \sum_{i=1}^n c_i\lambda_i\mathbf{Q}^{-1}\mathbf{u}_i = \sum_{i=1}^n c_i\lambda_i\mathbf{v}_i$. Таким образом, система собственных векторов $\{\mathbf{v}_i\}$ матрицы $\mathbf{A}$ полна.

6.23. а) $\mathbf{A} = b\mathbf{\Lambda} + (a+2b)\mathbf{E}$, $\quad \mathbf{\Lambda} = \begin{pmatrix} -2 & 1 & 0 & \dots & 0 \\ 1 & -2 & 1 & \dots & 0 \\ \vdots & \ddots & \ddots & \ddots & \vdots \\ 0 & \dots & 1 & -2 & 1 \\ 0 & \dots & 0 & 1 & -2 \end{pmatrix}$.

Матрица $\mathbf{\Lambda}$ имеет собственные векторы и собственные значения:

$$\mathbf{u}_k = \Big(\sin\frac{k\pi}{n+1}, \dots, \sin\frac{nk\pi}{n+1}\Big)^T,\ \lambda_k(\mathbf{\Lambda}) = -4\sin^2\frac{k\pi}{2(n+1)},\ k = 1, 2, \dots, n.$$

Матрица $\mathbf{A}$ имеет те же, что и Λ, собственные векторы $\mathbf{u}_k$ и собственные значения $\lambda_k(\mathbf{A}) = b\lambda_k(\Lambda) + a + 2b = a + 2b\cos\frac{k\pi}{n+1}$.

б) $\lambda_k(\mathbf{A}) = a + 2b\cos\frac{k\pi}{n+1} \geq a - 2|b|\cos\frac{\pi}{n+1}$.

Приложение А
Краткое введение в Матлаб

Матлаб – это одновременно современный язык программирования высокого уровня, пакет программ и среда для решения разнообразных научно-технических задач. Название Матлаб является сокращением от Matrix Laboratory и указывает на специальную ориентированность этого языка на действия с векторами и матрицами. Являясь интерактивной системой компьютерных вычислений, Матлаб дает прекрасные средства для обучения, научных исследований и решения практических задач. По сравнению с другими вычислительными средствами (Фортран, C и др.) Матлаб имеет ряд преимуществ:

• простота написания программ на языке высокого уровня;

• легкость работы со структурами данных, в частности, нет нужды описывать массивы перед их использованием;

• интерактивный интерфейс, обеспечивающий простоту проведения экспериментов и быструю отладку программ;

• наличие высококачественной графики и средств визуализации;

• M-файлы (программы на Матлабе) полностью портабельны для большого числа платформ;

• инструментальные средства Матлаба допускают расширение для включения, например, программ обработки сигналов, символьных вычислений и т. д.;

• наличие в интернете большого числа M-файлов, написанных различными пользователями Матлаба;

• возможность диалога с математическими системами Maple, Mathcad, MS Excel и др. расширяет средства Матлаба.

В результате указанных (и многих других) преимуществ Матлаб является одной из наиболее мощных математических систем, пользующейся большой популярностью у пользователей.

Следует однако отметить, что в Матлабе можно использовать только двумерные массивы. Матлаб работает как интерпретатор а не транслятор и, следовательно, программы на Матлабе выполняются существенно медленнее, чем программы на C, Фортране, Паскале и других языках, имеющих трансляторы. По этой причине Матлаб очень удобен для тестовых расчетов но «большие задачи математической физики» не программируют на Матлабе.

§ A.1. Начальные сведения

Наилучший способ научиться работать на Матлабе – пытаться экспериментировать. При этом следует иметь в виду следующее:

• войдя в Матлаб, вы можете печатать нужные вам команды сразу после указателя >> в командном окне;

• заглавные и строчные буквы в Матлабе различаются;

• печать имени переменной приводит к выводу на экран ее текущего значения;

• использование точки с запятой «;» в конце команды подавляет вывод результата на экран;

• Матлаб использует оба вида скобок ( ) и [ ], которые однако не являются взаимозаменяемыми;

• использование клавиш ↑ (*стрелочка вверх*) и ↓ (*стрелочка вниз*) позволяет прокручивать предыдущие команды;

• напечатав «help», вы получаете доступ к описанию команд, функций или символов;

• для выхода из Матлаба используются команды «exit» или «quit».

§ A.2. Операции над векторами

Матлаб трактует все переменные как матрицы. При этом векторы рассматриваются как одномерные матрицы. Для задания вектор-строки достаточно, например, напечатать:

```
>> a = [1 2 3]
```

В ответ вы получите на экране

```
a =
     1   2   3
```

Вектор-строку легко превратить в вектор-столбец с помощью операции транспонирования, используя знак апострофа,

```
>> a'
ans =
     1
     2
     3
```

Вектор-столбец может быть задан также следующим образом:

```
>> c = [4; 5; 6]
c =
```

4
5
6

Здесь точка с запятой указывает на переход к новой строке. Теперь можно перемножить два вектора a и c:

```
>> a*c
ans =
    32
```

Другой способ вычисления скалярного произведения векторов a и c:

```
>> dot(a,c)
ans =
    32
```

Произведение векторов c и a дает уже матрицу

```
>> A= c*a
A=
    4   8   12
    5   10  15
    6   12  18
```

Произведение $a * a$ не определено, так как размерности не совместимы для матричного умножения.

§ A.3. Два вида арифметических операций

В Матлабе имеется два вида арифметических операций над векторами и матрицами. *Матричные операции* производятся по обычным правилам линейной алгебры с использованием символов $+$, $-$, $*$, / и ^. *Поэлементные операции* выполняются покомпонентно с добавлением точки перед знаком такой операции. Таким образом, если мы хотим возвести в квадрат каждый элемент вектора a, то можно напечатать

```
>> b = a.^2
b =
    1   4   9
```

Так как векторы a и b имеют одинаковую длину, то можно найти их поэлементное произведение

```
>> a.*b
ans =
    1   8   27
```

В Матлабе имеется много математических функций, которые выполняются в поэлементном смысле, когда их аргумент является вектором или матрицей. Например,

```
>> exp(a)
ans =
    2,7183   7,3891   20,0855
>> log(ans)
ans =
    1   2   3
>> sqrt(a)
ans =
    1,0000   1,4142   1,7321
```

По умолчанию Матлаб выводит на экран числа с плавающей запятой с четырьмя десятичными разрядами, но при этом сами арифметические операции производятся с 14 десятичными разрядами. Для изменения формата вывода можно воспользоваться командой

```
>> format long
>> sqrt(a)
ans =
    1,00000000000000   1,41421356237310   1,73205080756888
>> format
```

Последняя команда восстанавливает используемый по умолчанию формат вывода на экран десятичных чисел. Большие и малые числа выводятся в экспоненциальной форме, где e предшествует степени десяти:

```
>> 2^(-24)
ans =
    5,9605e-008
```

Имеются также различные функции для операций над данными. Например, вычисление среднего значения и суммы элементов выполняется с помощью функций «mean» и «sum»:

```
>> mean(b), sum(c)
ans =
    5
ans =
    14
```

Этот пример показывает, что в строку может быть включено несколько команд, которые отделяются запятыми. Переменная pi закреплена за числом π.

```
>> pi
ans =
    3,1416
```

Если после команды стоит точка с запятой, то вывод результата на экран не производится:

```
>> y=tan(pi/6);
```

§ A.4. Операции над матрицами

Задание матрицы (двумерного массива) может быть выполнено следующим образом:

```
>> B = [-3 0 1; 2 5 -7; -1 4 8]
B =
    -3     0     1
     2     5    -7
    -1     4     8
```

Ядром Матлаба являются мощные средства решения задач линейной алгебры. Например, для решения системы линейных алгебраических уравнений $B * x = c$ достаточно использовать команду:

```
>> x = B\c
x =
    -1,3717
     1,3874
    -0,1152
```

Полученный результат можно проверить, вычислив евклидову норму невязки,

```
>> norm(B*x-c)
ans =
    0
```

Обратная матрица может быть найдена с помощью функции «inv»:

```
>> inv(B)
ans =
    -0,3560   -0,0209    0,0262
     0,0471    0,1204    0,0995
    -0,0681   -0,0628    0,0785
```

Собственные значения матрицы B можно найти, используя функцию «eig»:

```
>> e=eig(B)
e =
    -2,8601
```

```
    6,4300 + 5,0434i
    6,4300 − 5,0434i
```

Здесь i – мнимая единица, т. е. $i = \sqrt{-1}$. Результат работы функции «eig» можно также определить следующим образом:

```
>> [V,D] = eig(B)
V =
    0,9823    −0,0400 − 0,0404i    −0,0400 + 0,0404i
   −0,1275     0,7922               0,7922
    0,1374    −0,1733 − 0,5823i    −0,1733 + 0,5823i
D =
   −2,8601          0                    0
         0    6,4300 + 5,0434i           0
         0          0              6,4300 − 5,0434i
```

Здесь столбцы матрицы V являются собственными векторами матрицы B, а диагональные элементы матрицы D – соответствующими им собственными значениями.

Векторы с равноотстоящими значениями удобно строить, используя двоеточие:

```
>> v = 1:6
v =
    1    2    3    4    5    6
```

В общем случае m : n генерирует вектор с элементами $m, m+1, \ldots, n$. Отличный от единицы шаг может определить, используя конструкцию m : s : n, что дает элементы $m, m+s, \ldots, n$. Например,

```
>> w=2:3:10, y=1:−0,25:0
w =
    2    5    8
y =
    1,0000    0,7500    0,5000    0,2500    0
```

Матрицы можно расширять путем добавления столбцов и строк. Например, используя квадратные скобки, получаем

```
>> C = [A,[8; 9; 10] ], D = [B; a]

C =
    4    8   12    8
    5   10   15    9
    6   12   18   10
D =
```

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

Элемент в строке i и столбце j матрицы C, где счет по i и j всегда начинается с 1, может быть получен как C(i, j):

>> C(2, 3)

ans =

15

Конструкция C(i1 : i2, j1 : j2) позволяет вычленить подматрицу, стоящую на пересечении строк с i1 по i2 и столбцов с j1 по j2:

>> C(2:3,1:2)

ans =

$$\begin{matrix} 5 & 10 \\ 6 & 12 \end{matrix}$$

Некоторые матрицы строятся автоматически. Это относится, например, к единичной матрице и матрицам, состоящим только из нулей или единиц:

>> I3 = eye(3,3), Y = zeros(3,5), Z = ones(2)

I3 =

$$\begin{matrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{matrix}$$

Y =

$$\begin{matrix} 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \end{matrix}$$

Z =

$$\begin{matrix} 1 & 1 \\ 1 & 1 \end{matrix}$$

Отметим, что у приведенных функций первый аргумент определяет число строк матрицы а второй число ее столбцов. Если матрица является квадратной, то достаточно использовать один аргумент.

§ A.5. Некоторые полезные функции и циклы

Функции «rand» и «randn» генерируют случайные числа из равномерного распределения на отрезке $[0, 1]$ и соответственно нормального распределения на $(0, 1)$. Если требуется повторение эксперимента, то нужно задавать состояние этих двух генераторов случайных чисел. Ниже эти состояния фиксированы как 20:

```
>> rand('state',20), randn('state',20)
>> F = rand(3), G = randn(1,5)
F =
    0,7062   0,3586   0,8468
    0,5260   0,8488   0,3270
    0,2157   0,0426   0,5541
G =
    1,4051    1,1780   −1,1142    0,2474   −0,8168
```

Здесь одинарные кавычки действуют как ограничители строки, т. е. 'state' – это строка. Многие из функций Матлаба используют строчные аргументы.

В настоящий момент в нашем рабочем поле уже имеется достаточно много переменных. Список этих переменных можно получить, воспользовавшись командой «who»:

```
>> who
```

Это дает следующий список переменных:

```
A   F    Y    b   w
B   G    Z    c   x
C   I3   a    e   y
D   V    ans  v
```

Альтернативная команда «whos» дает более детальный список переменных с указанием их размера и класса.

Как и большинство языков программирования Матлаб имеет циклы. Например,

```
>> S = 0;
>> for i = 1:100, S = S + 1; end
>> S
S =
    100
```

Еще один вид циклов основывается на операторе «while», когда некоторая группа операторов выполняется до тех пор, пока условие остается верным.

```
>> S = 0;
>> while S < 100, S = S + 1; end
>> S
S =
    100
```

§ A.6. Графика

Для получения двумерных графиков используется функция «plot»:

>> x = 0:0.005:1; y = exp(10*x.*(x−1)).*sin(12*pi*x);

>> plot(x,y)

Здесь plot(x, y) по умолчанию соединяет точки x(i), y(i) сплошной линией. Матлаб открывает специальное окно, в которое выводится получаемая картинка. В нашем случае это рис. А.1. Чтобы наложить на эту картинку какой-либо дополнительный график, используется инструкция «hold on», отмена которой производится оператором «hold off». Полученную картинку можно подвергнуть различным преобразованиям, запомнить, используя один из употребительных форматов .jpg, .eps и др., а затем включить в доклад, отчет, статью и т. д.

Последовательность инструкций, написанных на Матлабе, полезно запомнить в некоторм файле для их последующего использования. Такой файл называется М-файлом. Допустим нам нужно отрисовать график уже рассмотренной функции $f(x) = e^{-10x(x-1)}\sin(12\pi x)$ для $x \in [0,1]$. Образуем М-файл, например, с именем «pict.m», помещаемый в текущую директорию.

Листинг М-файла pict.m

```
function pict(f,a,b)
x = a : 0.01 : b;
y = f1(x);
plot(x,y);
grid on;
hold on;
xlabel('x'); ylabel('f(x)');
title('Function graph');
function g = f(x)
g = exp(-10*x.*(x-1)).*sin(12*pi*x);
```

Теперь для отрисовки графика функции f на отрезке $[0,1]$ достаточно напечатать в командной строке:

>> pict('f',0,1)

Отрисовка поверхностей производится в Матлабе с помощью функций «mesh», «plot3» и «surf». Пусть требуется получить поверхность, описываемую функцией $z = e^{-x^2-y^2}$ на квадрате $[-4,4] \times [-4,4]$. Используя инструкции

>> [x,y] = meshgrid(−4,0 : 0,2 : 4,0,−4,0 : 0.2 : 4,0);

>> z=exp($-x.\hat{}2 - y.\hat{}2$);

>> mesh(x, y, z)

получаем поверхность, изображенную на рис. А.2.

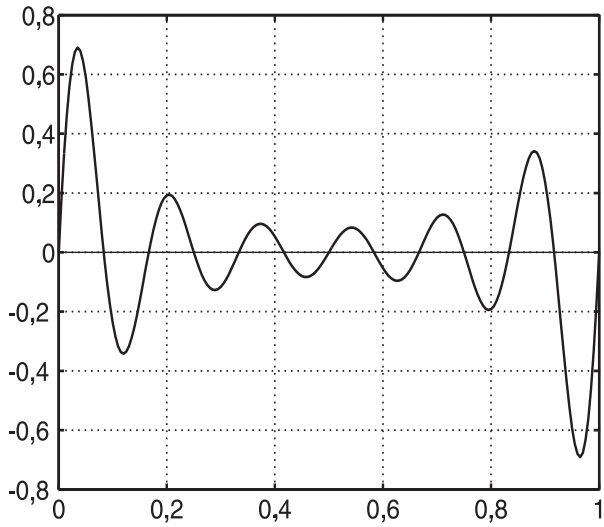


Рис. А.1. График функции $y = e^{10x(x-1)} \sin(12\pi x)$, $x \in [0, 1]$

В заключение приведем листинг М-файла «sweep.m», позволяющего строить поверхность вращения с помощью функции «surf». Здесь команда surf(X, Y, Z) создает трехмерную поверхность, где высота Z(i, j) задается в точке (X(i, j),Y(i, j)) плоскости xy. Символ процента % в листинге означает, что содержание строки, следующее за этим символом, является комментарием. Результирующая поверхность изображена на рис. А.3.

Листинг М-файла «sweep.m»

```
% sweep генерирует трехмерный объект вращения
N = 10; % число разбиений
z = linspace(-5,5,N);
radius = sqrt(1 + z.^2);
theta = 2*pi*linspace(0,1,N);
X = radius*cos(theta);
Y = radius*sin(theta);
Z = z(:,ones(1,N));
surf(X,Y,Z);
axis equal
```

Дальнейшие сведения о Матлабе могут быть получены из многочисленных руководств по этой системе и Интернета. В частности, при написании этого краткого введения автор использовал руководство [?].

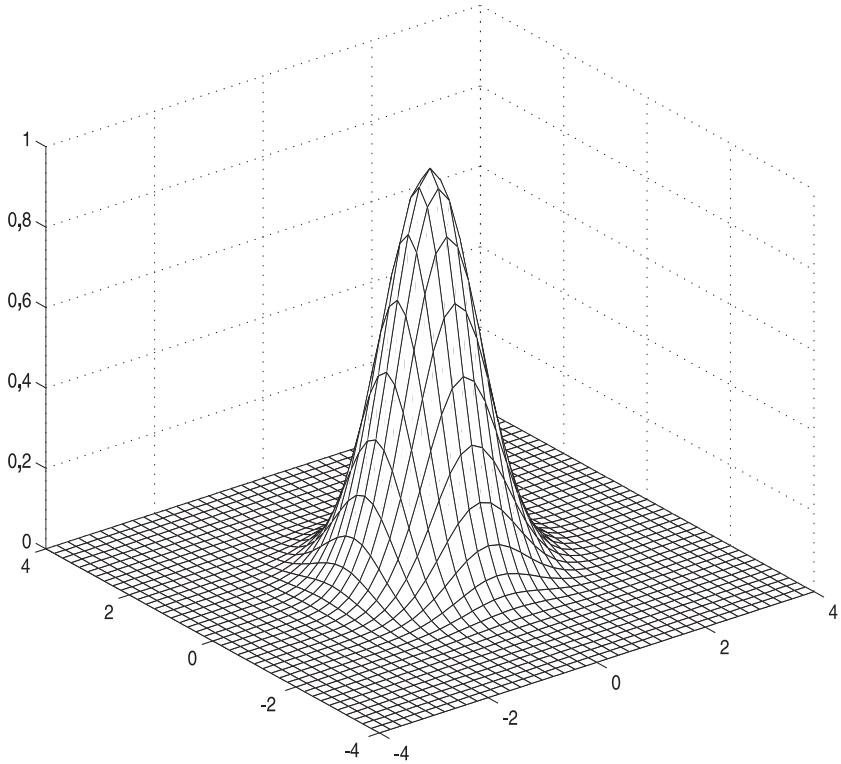


Рис. А.2. Трехмерная поверхность, полученная с помощью функции «mesh»

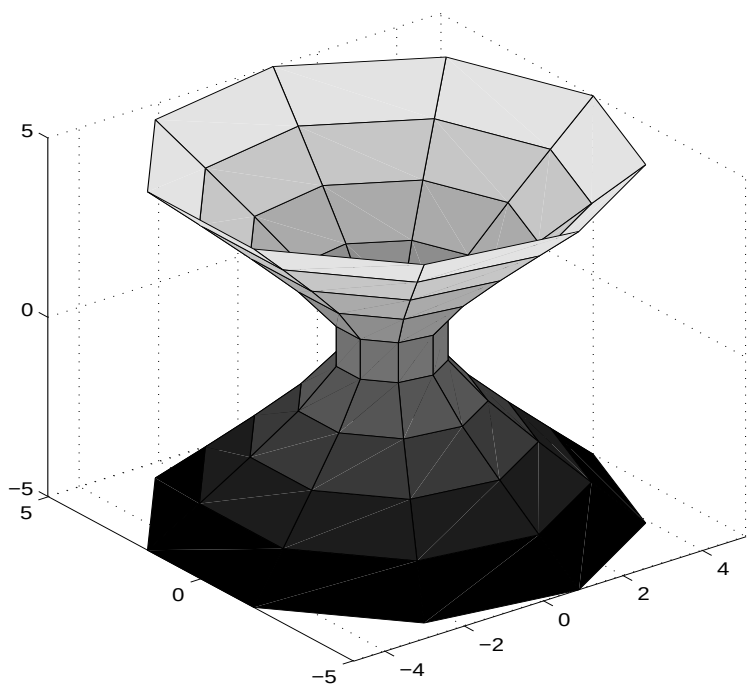


Рис. А.3. Поверхность вращения, полученная с помощью М-файла «sweep.m»

Приложение Б
Лабораторные работы

Лабораторный практикум в компьютерном классе является неотъемлемой частью курса современных методов вычислений. Для приводимых ниже лабораторных работ предполагается, что студент должен написать свои собственные программы на Матлабе, результаты работы которых ему было бы полезно сравнить с получаемыми по стандартным кодам Матлаба. При составлении вариантов заданий по компьютерному практикуму использовано учебное пособие [7].

Лабораторная работа № 1
Решение нелинейных уравнений

Постановка задачи. Требуется найти корни нелинейного уравнения

$$f(x) = 0 \quad \text{для} \quad x \in [a, b],$$

где f – непрерывная или более гладкая функция.

Описание метода. Решение задачи состоит из двух этапов:

а) отделение корней;

б) уточнение корней.

Первый этап предлагается выполнить аналитически и/или графически, используя компьютер. На втором этапе предполагается, что $f(a)f(b) < 0$ и функция f имеет на $[a, b]$ единственный корень. Для его уточнения с нужным числом правильных знаков используются два итерационных метода. Это могут быть, например, метод проб и метод Ньютона, которые предполагается сравнить. В обоих случаях итерации прекращаются, когда для приближений к корню $\{x_k\}$ имеем $|x_{k+1} - x_k| \le \varepsilon$, где $\varepsilon > 0$ – заданное малое число.

Задание. Выполнение задания состоит из четырех частей:

1. отделить корни аналитически;

2. отделить корни аналитически и уточнить один из них до 0,01 методом проб;

3. отделить корни и уточнить один из них до 0,01 графически;

4. отделить корни графически и уточнить один из них до 0,01 двумя итерационными методами, сравнить использованные методы.

Варианты.

№ 1. 1) $2^x + 5x - 3 = 0$;
2) $3x^4 + 4x^3 - 12x^2 - 5 = 0$;
3) $0{,}5^x + 1 - (x-2)^2 = 0$;
4) $(x-3)\cos x - 1 = 0;\ |x| \le 2\pi$.

№ 2. 1) $\operatorname{arctg}(x) - \frac{1}{3x^3} = 0$;
2) $2x^3 - 9x^2 - 60x + 1 = 0$;
3) $[\log_2(-x)] \cdot (x+2) = -1$;
4) $\sin\left(x + \frac{\pi}{3}\right) - 0{,}5x = 0$.

№ 3. 1) $5^x + 3x = 0$;
2) $x^4 - x - 1 = 0$;
3) $x^2 - 2 + 0{,}5^x = 0$;
4) $(x-1)^2 \cdot \lg(x+11) = 1$.

№ 4. 1) $2e^x = 5x + 2$;
2) $2x^4 - x^2 - 10 = 0$;
3) $x \cdot \log_3(x+1) = 1$;
4) $\cos(x + 0{,}5) = x^3$.

№ 5. 1) $3^{x-1} - 2 - x = 0$;
2) $3x^4 + 8x^3 + 6x^2 - 10 = 0$;
3) $(x-4)^2 \cdot \log_{0{,}5}(x-3) = -1$;
4) $5\sin x = x$.

№ 6. 1) $2\operatorname{arctg}(x) - \frac{1}{2x^3} = 0$;
2) $x^4 - 18x^2 + 6 = 0$;
3) $x^2 \cdot 2^x = 1$;
4) $\operatorname{tg}(x) = x + 1,\ |x| \le \pi/2$.

№ 7. 1) $e^{-2x} - 2x + 1 = 0$;
2) $x^4 + 4x^3 - 8x^2 - 17 = 0$;
3) $0{,}5^x - 1 = (x+2)^2$;
4) $x^2 \cos 2x = -1$.

№ 8. 1) $5^x - 6x - 3 = 0$;
2) $x^4 - x^3 - 2x^2 + 3x - 3 = 0$;
3) $2x^2 - 0{,}5^x - 3 = 0$;
4) $x\lg(x+1) = 1$.

№ 9. 1) $\operatorname{arctg}(x-1) + 2x = 0$;
2) $3x^4 + 4x^3 - 12x^2 + 1 = 0$;
3) $(x-2)^2 2^x = 1$;
4) $x^2 - 20\sin x = 0$.

№ 10. 1) $2\operatorname{arctg}(x) - x + 3 = 0$;
2) $3x^4 - 8x^3 - 18x^2 + 2 = 0$;
3) $2\sin\left(x + \frac{\pi}{3}\right) = 0{,}5x^2 - 1$;
4) $2\lg x - \frac{x}{2} + 1 = 0$.

№ 11. 1) $3^x + 2x - 2 = 0$;
2) $2x^4 - 8x^3 + 8x^2 - 1 = 0$;
3) $[(x-2)^2 - 1]2^x = 1$;
4) $(x-2)\cos x = 1,\ |x| \le 2\pi$.

№ 12. 1) $2\operatorname{arctg}(x) - 3x + 2 = 0$;
2) $2x^4 + 8x^3 + 8x^2 - 1 = 0$;
3) $[\log_2(x+2)](x-1) = 1$;
4) $\sin(x - 0{,}5) - x + 0{,}8 = 0$.

№ 13. 1) $3^x + 2x - 5 = 0$;
2) $x^4 - 4x^3 - 8x^2 + 1 = 0$;
3) $x^2 - 3 + 0{,}5^x = 0$;
4) $(x-2)^2 \lg(x+11) = 1$.

№ 14. 1) $2e^x + 3x + 1 = 0$;
2) $3x^4 + 4x^3 - 12x^2 - 5 = 0$;
3) $x\log_3(x+1) = 2$;
4) $\cos(x + 0{,}3) = x^2$.

№ 15. 1) $3^{x-1} - 4 - x = 0$;
2) $2x^3 - 9x^2 - 60x + 1 = 0$;
3) $(x-3)^2 \log_{0{,}5}(x-2) = -1$;
4) $5\sin x = x - 1$.

№ 16. 1) $\operatorname{arctg}(x) - \frac{1}{3x^3} = 0$;
2) $x^4 - x - 1 = 0$;
3) $(x-1)^2 2^x = 1$;
4) $\operatorname{tg}^3 x = x - 1,\ |x| \le \pi/2$.

Решение варианта № 1.

1. Рассмотрим уравнение $f(x) = 2^x + 5x - 3 = 0$. Для выяснения характера поведения функции f продифференцируем ее и приравняем производную нулю. Здесь $f'(x) = 2^x \ln 2 + 5 = 0$, т. е. $2^x = -5/\ln 2$. Поскольку $\ln 2 > 0$, то это уравнение не имеет решения и $f'(x) > 0$. Следовательно, функция f монотонно возрастает. Так как $f(-\infty) < 0$ и $f(\infty) > 0$, то функция f имеет единственный корень. Рассмотрим несколько характерных точек. Представим их в виде таблицы:

x	$-\infty$	-1	0	1	2	$+\infty$
$sign(f(x))$	$-$	$-$	$-$	$+$	$+$	$+$

Ответ. Функция f меняет знак на отрезке $[0, 1]$ и имеет здесь один корень.

2. Рассмотрим уравнение $f(x) = 3x^4 + 4x^3 - 12x^2 - 5 = 0$. Дифференцируя, находим, что $f'(x) = 12x^3 + 12x^2 - 24x = 12(x + 2)x(x - 1)$. Таким образом, производная $f'(x) = 0$ в трех точках $x = -2$, $x = 0$ и $x = 1$. Так как $f(-\infty) > 0$ и $f(\infty) > 0$, то функция f имеет в точке $x = -2$ минимум, в $x = 0$ – максимум и в $x = 1$ – минимум. На отрезках $[-\infty, -2]$, $[-2, 0]$, $[0, 1]$ и $[1, \infty]$ функция f монотонна и имеет не более одного корня. Чтобы уточнить положение корней, рассмотрим несколько характерных точек, включая точки экстремума. Так как $f(-3) = 22$, $f(-2) = -37$, $f(0) = -5$, $f(1) = -10$ и $f(2) = 27$, то функция f имеет два корня, которые находятся соответственно на отрезках $[-3, -2]$ и $[1, 2]$.

Уточним теперь положение корня на отрезке $[1, 2]$ с точностью до 0,01. Используем метод проб. С этой целью составим таблицу:

iter	a	b	c	$f(c)$	$\lvert b-a\rvert/2$
0	1,0000	2,0000	1,5000	−3,3125	0,5000
1	1,5000	2,0000	1,7500	7,8242	0,2500
2	1,5000	1,7500	1,6250	1,3952	0,1250
3	1,5000	1,6250	1,5625	−1,1567	0,0625
4	1,5625	1,6250	1,5937	0,0677	0,0312
5	1,5625	1,5938	1,5781	−0,5571	0,0156
6	1,5781	1,5938	1,5859	−0,2479	0,0078

Ответ. Функция f на отрезке $[1, 2]$ имеет единственный корень $x \approx 1{,}58$.

3. Для отрисовки графика функции $f(x) = 0{,}5^x + 1 - (x - 2)^2$ воспользуемся М-файлом «pict.m». Это дает нам график на рис. Л1.1. Нетрудно видеть, что функция f имеет корень вблизи точки $x = -6$ и возможно на отрезке $[0, 4]$. Повторным применением функции «pict» уточним наличие корней на отрезке $[0, 4]$. На рис. Л1.2 а) видно, что в действительности здесь имеется два корня, причем согласно рис. Л1.2 б) второй из них с точностью до 0,01 равен 3,058.

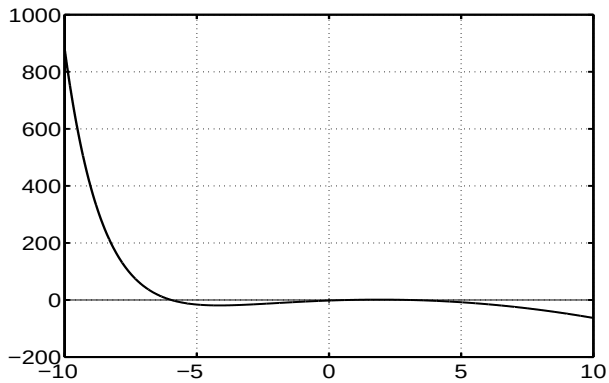


Рис. Л1.1. Отделение корней функции $f(x) = (x-3)\cos x - 1$ графически

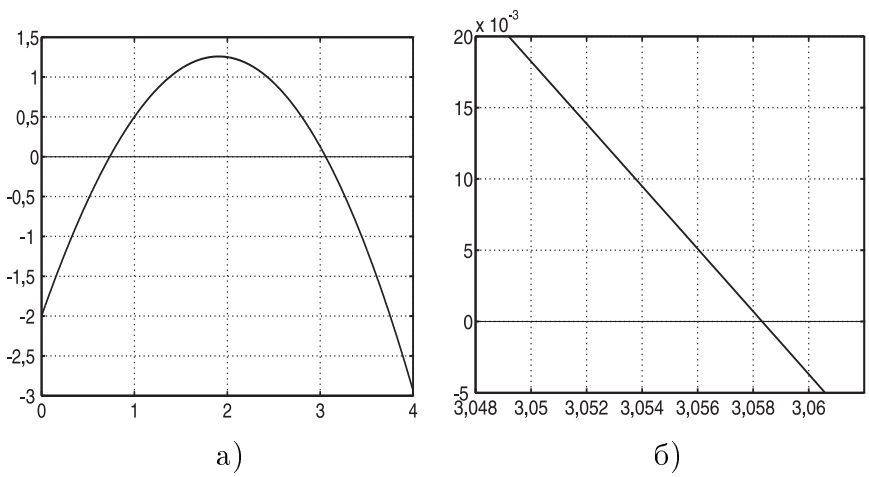


Рис. Л1.2. Отделение корней функции $f(x) = (x-3)\cos x - 1$ графически: а) на отрезке $[0; 4]$; б) на отрезке $[3,05; 3,06]$

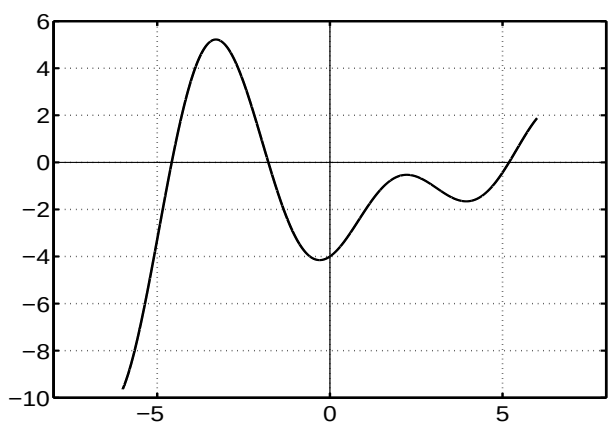


Рис. Л1.3. Отделение корней функции $f(x) = (x-3)\cos x - 1$ графически

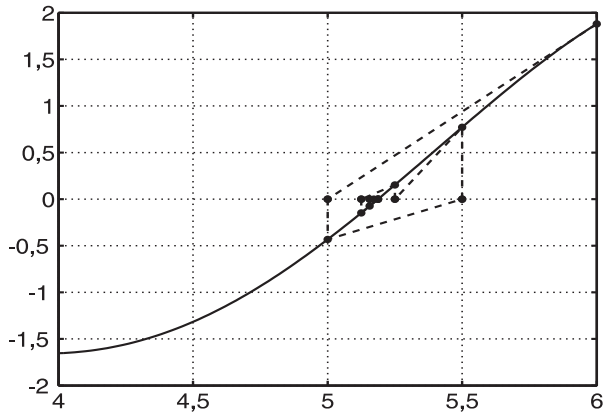


Рис. Л1.4. Уточнение корня функции $f(x) = (x-3)\cos x - 1$ методом проб

Ответ. Значения корней: $x_1 \approx -5{,}965$, $x_2 \approx 0{,}735$ и $x_3 \approx 3{,}058$.

4. Для отыскания корней функции $f(x) = (x-3)\cos x - 1$, $-2\pi \le x \le 2\pi$ предварительно локализуем их, используя опять М-файл «pict.m». Согласно рис. Л1.3 на отрезке $[-2\pi, 2\pi]$ эта функция имеет три корня. Уточним корень, находящийся на отрезке $[4, 6]$. Вначале используем метод проб. На рис. Л1.4 пунктиром показана сходимость итераций по этому методу.

Метод проб дает следующую таблицу: >> bisect('f', 4, 6, 0.01, 15)

iter	a	b	c	$f(c)$	$\lvert b-a\rvert/2$
0	4,0000	6,0000	5,0000	−0,4327	1,0000
1	5,0000	6,0000	5,5000	0,7717	0,5000
2	5,0000	5,5000	5,2500	0,1522	0,2500
3	5,0000	5,2500	5,1250	−0,1479	0,1250
4	5,1250	5,2500	5,1875	0,0006	0,0625
5	5,1250	5,1875	5,1562	−0,0740	0,0312
6	5,1563	5,1875	5,1719	−0,0368	0,0156
7	5,1719	5,1875	5,1797	−0,0181	0,0078

По методу Ньютона получаем: >> newton('f', 'df', 6, 0.01, 15)

iter	x_n	$f(x_n)$	$f'(x_n)$	$\lvert x_{n+1} - x_n\rvert$
0	6,0000	1,8805	1,7984	
1	4,9543	−0,5317	2,1370	1,0456
2	5,2032	0,0384	2,4144	0,2488
3	5,1873	0,0001	2,4025	0,0159
4	5,1872	0,0000	2,4024	0,0000

Листинг M-файла «bisect.m»

```
function bisect(f,a,b,tol,n)
% Метод проб для решения нелинейного уравнения
%            f(x)=0,  a <= x <= b.
% tol - требуемая точность, n - допустимое число итераций.
h=(b-a)/100;
x=a:h:b;
y=feval(f,x);
plot(x,y,'-k','LineWidth',2);
hold on;
grid on;
iter=0;
u=feval(f,a); v=feval(f,b); c=(a+b)*0.5;
err=abs(b-a)*0.5; xx(1)=b; xx(2)=c; yy(1)=feval(f,b); yy(2)=0;
plot(xx,yy,'-ok','Linewidth',2,'MarkerSize',5,'MarkerFaceColor','k');
      disp('------------------------------------------------------------')
      disp('  iter       a        b       c       f(c)       |b-a|/2')
      disp('------------------------------------------------------------')
fprintf('\n')
if(u*v<=0)
while (err>tol) & (iter<=n)
      w=feval(f,c); p=c;
fprintf('%2.0f%10.4f%10.4f%10.4f%10.4f%10.4f\n',iter,a,b,c,w,err)
      if(w*u<0) b=c; v=w; end;
      if(w*u>0) a=c; u=w; end;
      iter=iter+1;  c=(a+b)*0.5; err=abs(b-a)*0.5;
      if(p<c) p=a; else p=b; end
      xx(1)=p; xx(2)=p; xx(3)=c; yy(1)=0; yy(2)=feval(f,p); yy(3)=0;
plot(xx,yy,'-ok','Linewidth',2,'MarkerSize',5,'MarkerFaceColor','k');
end; w=feval(f,c);
fprintf('%2.0f%10.4f%10.4f%10.4f%10.4f%10.4f\n',iter,a,b,c,w,err)
if(iter>n)
disp('Метод не сходится'); end;
else  disp('Метод не может быть использован: f(a)f(b)>0'); end;
      disp('------------------------------------------------');

function g=f(x)
g=(x-3).*cos(x)-1;
```

Лабораторная работа № 2

Интерполяция

Постановка задачи. Пусть имеются некоторые данные (x_i, f_i); $i = 0, 1, \ldots, N$, где $a = x_0 < x_1 < \ldots < x_N = b$. Требуется интерполировать эти данные, используя интерполяционный многочлен Лагранжа и кубический сплайн.

Описание метода. Для интерполяции используется многочлен Лагранжа в форме Ньютона

$$L_{N+1}(x) = c_0 + c_1(x - x_0) + \ldots + c_N(x - x_0)\ldots(x - x_{N-1}),$$

где

$$\begin{aligned} c_0 &= f[x_0] \equiv f_0, \\ c_k &= \frac{f[x_1, \ldots, x_k] - f[x_0, \ldots, x_{k-1}]}{x_k - x_0}, \quad k = 1, 2, \ldots, N. \end{aligned}$$

Вначале вычисляются разделенные разности c_k а затем значения интерполяционного многочлена L_{N+1} находятся по схеме Горнера

$$L_{N+1}(x) = c_0 + (x - x_0)(c_1 + \ldots + (x - x_{N-2})(c_{N-1} + (x - x_{N-1})c_N)\ldots).$$

Для построения интерполяционного кубического сплайна S решается система линейных уравнений с трехдиагональной матрицей

$$\begin{aligned} 2M_0 + M_1 &= \frac{6}{h_0}(f[x_0, x_1] - f'_0), \\ \frac{h_{i-1}}{h_{i-1} + h_i} M_{i-1} + 2M_i + \frac{h_i}{h_{i-1} + h_i} M_{i+1} &= 6f[x_{i-1}, x_i, x_{i+1}], \\ & \quad i = 1, 2, \ldots, N-1, \\ M_{N-1} + 2M_N &= \frac{6}{h_{N-1}}(f'_N - f[x_{N-1}, x_N]), \end{aligned}$$

где $M_i = S''(x_i)$. Значения сплайна на $[x_i, x_{i+1}]$ вычисляются по формуле

$$S(x) = f_i(1 - t) + f_{i+1}t - t(1 - t)\frac{h_i^2}{6}[(2 - t)M_i + (1 + t)M_{i+1}],$$

где $t = (x - x_i)/h_i$.

Задание.

1. Используя значения функции $f_i = f(x_i)$ на равномерной сетке $x_i = a + ih$, $h = (b - a)/N$, образовать таблицу исходных данных (x_i, f_i), $i = 0, 1, \ldots, N$.

2. Написать программы на Матлабе, реализующие интерполяцию многочленом Лагранжа и кубическим сплайном.

3. Для тестирования программ в качестве функции f использовать кубический многочлен.

4. Построить интерполяционные кривые многочлена Лагранжа и кубического сплайна и сравнить их с графиком исходной функции.

5. Рассмотреть поведение многочлена Лагранжа и кубического сплайна при увеличении числа узлов интерполяции.

Варианты.

№ 1. $f(x) = e^x + x + 1$;
№ 2. $f(x) = 2x^4 - x^2 - 10$;
№ 3. $f(x) = 0{,}5^x - 3 - (x+2)^2$;
№ 4. $f(x) = x^2\cos(2x) + 1$;
№ 5. $f(x) = 3^x - 2x + 5$;
№ 6. $f(x) = 3x^4 + 8x^3 + 6x^2 - 10$;
№ 7. $f(x) = 2x^2 - 0{,}5^x - 2$;
№ 8. $f(x) = x\lg(x+1) - 1$;
№ 9. $f(x) = \mathrm{arctg}(x-1) + 3x - 2$;
№ 10. $f(x) = x^4 - 18x^3 + 6$;
№ 11. $f(x) = (x-2)^2 2^x - 1$;
№ 12. $f(x) = x^2 - 20\sin x$;
№ 13. $f(x) = 2\mathrm{arctg}(x) - x + 3$;
№ 14. $f(x) = x^4 + 4x^4 3 - 8x^2 - 17$;
№ 15. $f(x) = 2\sin(x + \pi/3) - x^2 + 0{,}5$;
№ 16. $f(x) = 2\lg(x) - x/2 + 1$.

Образец.

1. Правильность работы программ интерполяции тестировалась на кубическом многочлене $f(x) = 4x^3 - 12x^2 - 5$.

2. Интерполируемая функция: $f(x) = 1/(1 + 25x^2), \quad |x| \le 1$.

Решение. Вычислим значения функции f для $x_i = ih$; $h = 1/N$, $i = 0, 1, \ldots, N$ при $N = 4$ и $N = 8$ и интерполируем эти данные многочленом Лагранжа и кубическим сплайном. На рис. Л2.1 и Л2.2 сплошной, пунктирной и штрих-пунктирной линиями показаны соответственно графики функции f и интерполяционных многочленов Лагранжа и кубических сплайнов при пяти и девяти узлах интерполяции. Нетрудно видеть, что увеличение числа узлов интерполяции ухудшает качество приближения многочленом Лагранжа. Для кубического сплайна напротив имеет место сходимость его к интерполируемой функции. Таким образом, кубические сплайны являются существенно лучшим аппаратом приближения по сравнению с многочленами Лагранжа.

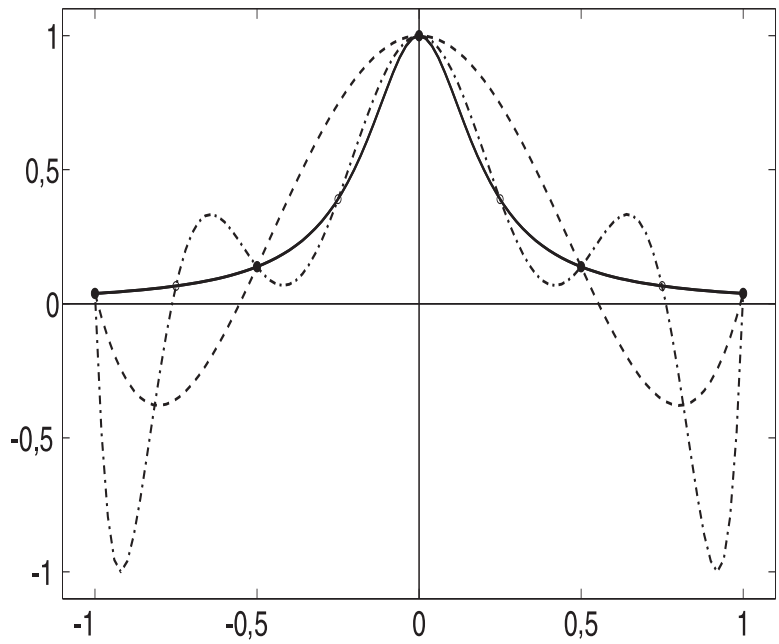


Рис. Л2.1. Интерполяция многочленами Лагранжа. Сплошная линия – график интерполируемой функции с отмеченными на нем точками исходных данных. Пунктирной и штрих-пунктирной линиями показаны графики интерполяционных многочленов Лагранжа 4-й и 8-й степени

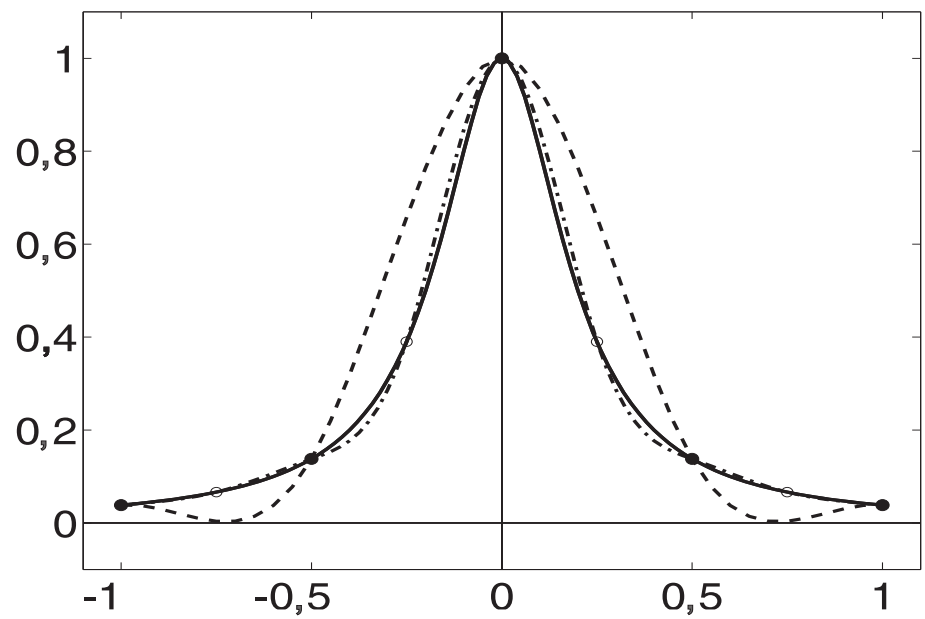


Рис. Л2.2. Интерполяция кубическими сплайнами. Сплошная линия – график интерполируемой функции с отмеченными на нем точками исходных данных. Пунктирной и штрих-пунктирной линиями показаны графики интерполяцион- ного кубического сплайна при пяти и девяти узлах интерполяции

Лабораторная работа № 3

Метод наименьших квадратов

Постановка задачи. Пусть имеются «экспериментальные» данные (x_i, y_i); $i = 0, 1, \ldots, N$, где $a = x_0 < x_1 < \ldots < x_N = b$, и задана система линейно независимых на $[a, b]$ легко вычислимых функций φ_j, $j = 1, 2, \ldots, M$. Требуется найти функцию $S(x) = \sum_{j=1}^{M} c_j \varphi_j(x)$ такую, что среднеквадратическое уклонение

$$I(c_1, \ldots, c_M) = \sum_{i=0}^{N} (S(x_i) - y_i)^2 = \sum_{i=0}^{N} \Big(\sum_{j=1}^{M} c_j \varphi_j(x_i) - y_i \Big)^2$$

достигает минимума.

Описание метода. Используя необходимое условие экстремума $\partial I / \partial c_k = 0$; $k = 1, 2, \ldots, M$, получаем систему M линейных алгебраических уравнений для нахождения M неизвестных коэффициентов c_j:

$$\sum_{j=1}^{M} \Big(\sum_{i=0}^{N} \varphi_j(x_i) \varphi_k(x_i) \Big) c_j = \sum_{i=0}^{N} y_i \varphi_k(x_i), \quad k = 1, 2, \ldots, M,$$

называемую нормальной системой метода наименьших квадратов (МНК). Так как функции φ_j, $j = 1, 2, \ldots, M$ по условию линейно независимы на $[a, b]$, то определитель нормальной системы будет отличен от нуля и ее решение может быть получено, например, методом исключения Гаусса.

Задание.

1. Используя значения функции $y_i = f(x_i)$ на равномерной сетке $x_i = a + ih$, $h = (b - a)/N$, образовать таблицу исходных данных (x_i, y_i), $i = 0, 1, \ldots, N$.

2. Написать программу на Матлабе, реализующую МНК.

3. В качестве базисных функций φ_j, $j = 1, 2, \ldots, M$ рассмотреть:

   а) мономы x^{j-1};

   б) базисные сплайны $B_{k,j}$, $k = 2, 4$ (см. гл. 3).

4. Для тестирования программы в качестве функции f использовать многочлен степени $k - 1 \leq 3$.

5. Аппроксимировать данные (x_i, y_i), $i = 0, 1, \ldots, N$ по МНК и сравнить графики функций f и S.

Варианты.

№ 1. $f(x) = x - \sin(x) - 0{,}25$.

№ 2. $f(x) = \operatorname{tg}(0{,}58x + 0{,}1) - x^2$.

№ 3. $f(x) = \sqrt{x} - \cos(0{,}387x)$.

№ 4. $f(x) = \operatorname{tg}(0{,}4x + 0{,}4) - x^2$.

№ 5. $f(x) = \lg(x) - 7/(2x + 6)$.

№ 6. $f(x) = \mathrm{tg}(0{,}5x + 0{,}2) - x^2$.

№ 7. $f(x) = 3x - \cos(x) - 1$.

№ 8. $f(x) = x + \lg(x) - 0{,}5$.

№ 9. $f(x) = \mathrm{tg}(0{,}5x + 0{,}1) - x^2$.

№ 10. $f(x) = x^2 + 4\sin(x)$.

№ 11. $f(x) = \mathrm{ctg}(1{,}05x) - x^2$.

№ 12. $f(x) = \mathrm{tg}(0{,}4x + 0{,}3) - x^2$.

№ 13. $f(x) = x\lg(x) - 1{,}2$.

№ 14. $f(x) = 1{,}8x^2 - \sin(10x)$.

№ 15. $f(x) = \mathrm{ctg}(x) - x/4$.

№ 16. $f(x) = \mathrm{tg}(0{,}3x + 0{,}4) - x^2$.

Образец.

1. Кубический многочлен для тестировки правильности работы программы МНК: $f(x) = 4x^3 - 12x^2 - 5$.

2. Функция для получения исходных данных:
$f(x) = [0{,}1 + (x - 0{,}2)^2]^{-1} + [0{,}15 + (x - 0{,}8)^2]^{-1}; \quad 0 \le x \le 1.$

Решение. Вычислим значения функции f для $x_i = ih$; $h = 1/N$, $i = 0, 1, \ldots, N$ при $N = 10$ и аппроксимируем эти данные по МНК, используя три вида базисных функций: мономы, B-сплайны первой и третьей степени. На рис. Л3.1 и Л3.2 пунктирной, штрих-пунктирной и точечной линиями показаны соответственно графики полученных по МНК аппроксимирующих многочленов, ломаных и кубических сплайнов степени (с числом звеньев) $M = 1, 3, 10$. На всех рисунках сплошной линией показан график аппроксимируемой функции f с отмеченными на нем черными кружками исходными данными. Очевидна сходимость МНК приближений при увеличении числа базисных функций к исходной функции f.

Для сравнения результатов полезно воспользоваться стандартными средствами Матлаба. Функция $p = polyfit(x, y, m)$ находит коэффициенты МНК многочлена p степени m для данных x, y. Функция $y = polyval(p, x)$ вычисляет значение многочлена p степени m в точке x. Аппроксимацию через B-сплайны можно получить с помощью функции $spap2(knots, k, x, y)$, где $knots$ – массив узлов сплайна, k – порядок сплайна а x, y – исходные данные.

Лабораторная работа № 4
Сглаживание кубическими сплайнами

Постановка задачи. Пусть имеются «экспериментальные» данные (x_i, z_i), $i = 0, 1, \ldots, N$, где $a = x_0 < x_1 < \ldots < x_N = b$. Требуется построить гладкую

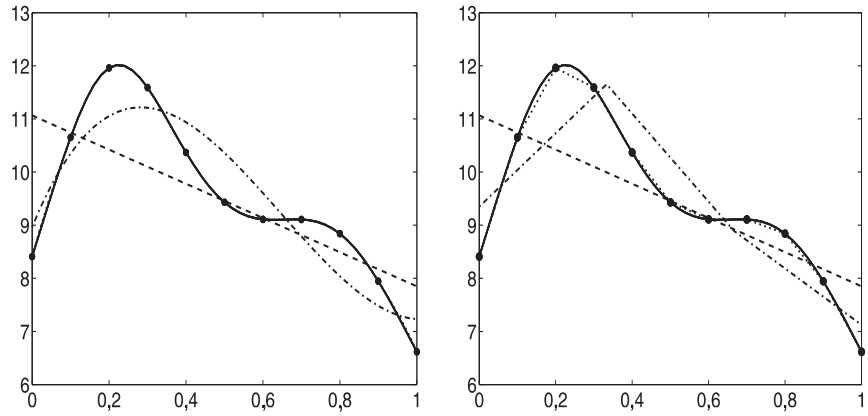


Рис. Л3.1. Графики, полученные по МНК: а) многочлены и б) ломаные степени (с числом звеньев) $M = 1, 3, 10$. Сплошная линия – график аппроксимируемой функции с отмеченными на нем черными точками исходными данными

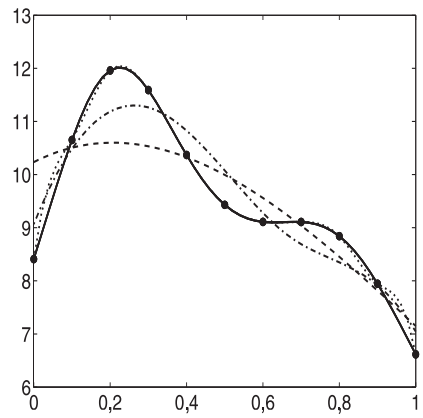


Рис. Л3.2. Графики полученных по МНК кубических сплайнов с разным числом звеньев $M = 1, 3, 10$. Сплошная линия – график аппроксимируемой функции с отмеченными на нем черными точками исходными данными

кривую, которая «сглаживала» бы эти данные, т. е. эта кривая была бы плавной и проходила бы достаточно близко к данным.

Описание метода. Искомую кривую ищем в виде кубического сплайна S, который доставляет минимум квадратическому функционалу

$$J_\alpha(S) = \sum_{i=0}^{N} p_i[S(x_i) - z_i]^2 + \alpha \int_a^b [S''(x)]^2 dx.$$

При $\alpha = 0$ имеем интерполяционный кубический сплайн. Для больших значений α сплайн стремится к прямой линии. Промежуточные значения α дают сглаживание. Для построения кубического сплайна методом пятиточечной прогонки решается система (3.23) и используются формулы (3.27)–(3.28). Если нет других соображений, то полагаем веса $p_i = 1$ для всех i.

Задание.

1. Используя значения функции $y_i = f(x_i)$ на равномерной сетке $x_i = a + ih$, $h = (b-a)/N$, образовать таблицу исходных данных (x_i, y_i), $i = 0, 1, \ldots, N$.

2. Получить зашумленные на 20% данные (x_i, z_i), $i = 0, 1, \ldots, N$, используя датчик случайных чисел.

3. Написать программу на Матлабе, реализующую сглаживание кубическими сплайнами.

4. Задать α. Выбрать весовые множители $p_i = 1$, $i = 0, 1, \ldots, N$, а затем уточнить их по формулам (3.37)–(3.38).

5. Параметр сглаживания α подобрать экспериментально, рассмотрев случаи переглаживания, оптимального выбора параметра сглаживания и интерполяции кубическим сплайном.

Варианты.

№ 1. $f(x) = \sin^2 \pi x, \quad 0 \le x \le 1.$

№ 2. $f(x) = (e^{1-x^2} - 1)/(e-1), \quad |x| \le 1.$

№ 3. $f(x) = \cos(\pi x), \quad 0 \le x \le 1.$

№ 4. $f(x) = \sin(2\pi x), \quad 0 \le x \le 1.$

№ 5. $f(x) = 1 - (e^{2x} - 1)/(e^2 - 1), \quad 0 \le x \le 1.$

№ 6. $f(x) = \cos^2(\pi x), \quad |x| \le 1/2.$

№ 7. $f(x) = \cos(\pi x), \quad 0 \le x \le 1.$

№ 8. $f(x) = \sin(2\pi x), \quad |x| \le 1/2.$

№ 9. $f(x) = x(1-x), \quad 0 \le x \le 1.$

№ 10. $f(x) = x^2(1-x)^2, \quad 0 \le x \le 1.$

№ 11. $f(x) = \begin{cases} 2/3 - x^2 + |x|^3/2, & |x| \le 1, \\ (2-|x|)^3/6, & 1 \le |x| \le 2, \\ 0, & 2 \le |x|. \end{cases}$

№ 12. $f(x) = \sqrt{x}, \quad 0 \le x \le 1.$

№ 13. $f(x) = e^x, \quad 0 \le x \le 1.$

№ 14. $f(x) = \ln x, \quad 1 \le x \le 2.$

№ 15. $f(x) = 1/x, \quad 1 \le x \le 2.$

№ 16. $f(x) = 1 - x^4, \quad |x| \le 1.$

Решение варианта № 1.

В качестве функции для получения экспериментальных данных возьмем функцию $f(x) = \sin^2(\pi x)$, $0 \le x \le 1$. Получение нужных зашумленных данных осуществим использованием генератора случайных чисел. Использование генератора случайных чисел с фиксированным состоянием позволяет обеспечить повторяемость зашумленных значений.

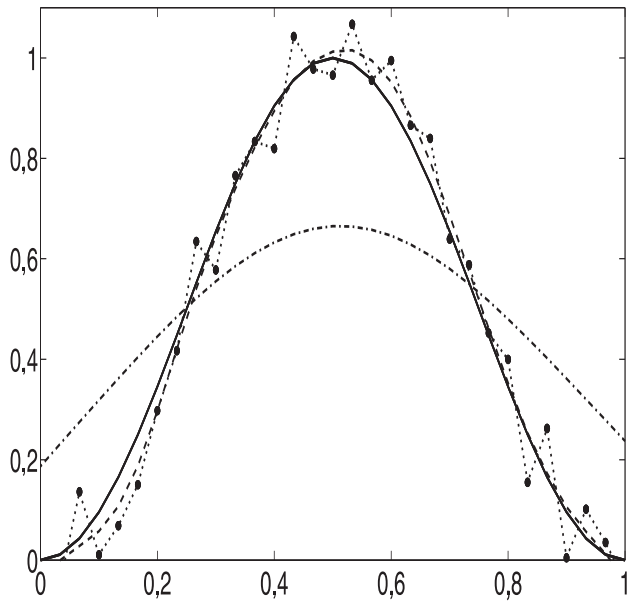


Рис. Л4.1. Сглаживание кубическим сплайном выборки значений функции-шапочки $f(x) = \sin^2(\pi x)$, $0 \le x \le 1$, на равномерной сетке с $N = 30$ при 20% шуме

Листинг M-файла получения зашумленных данных

```
function data(n,alpha);
rand('state',10);
for i=1:n
      x(i)=(i-1)/(n-1);
      y(i)=sin(pi*x(i)).^2;
      z(i)=y(i)+0.2*(rand-0.5)
end
```

Результаты работы программы smoothing(x,y,z,n,alpha), реализующей построение сглаживающего кубического сплайна даны на рис. Л4.1. График исходной функции $f(x) = \sin^2(\pi x)$ показан сплошной линией. Черными точками отмечены «экспериментальные» данные. Штрих-пунктирной, пунктирной и точечной линиями обозначены графики сглаживающего сплайна при значениях параметра сглаживания $\alpha = 10^{-k}$; $k = 1, 4, 8$. Случай $\alpha = 0,1$ соответствует типичному «переглаживанию». При «оптимальном» $\alpha = 0,0001$ сплайновая кривая близка к графику функции f но при этом она строится, исходя из зашумленных значений. Поэтому мы не можем получить точного восстановления исходной функции. Наконец, при $\alpha = 10^{-8}$ сплайн фактически интерполирует зашумленные данные и поэтому дает «осциллирующую» кривую.

Лабораторная работа № 5
Численное интегрирование

Постановка задачи. Требуется приближенно вычислить значение определенного интеграла

$$I(f) = \int_a^b f(x) d\, x,$$

где функция f не имеет особенностей на конечном отрезке $[a, b]$.

Описание метода. Отрезок интегрирования $[a, b]$ разобьем на N равных частей точками $x_i = a + ih$, $i = 0, 1, \ldots, N$, где $h = (b - a)/N$. Для приближенного вычисления интеграла $I(f)$ воспользуемся составными формулами трапеций

$$I(f) \approx \frac{h}{2} \sum_{i=0}^{N-1} [f(x_i) + f(x_{i+1})]$$

и Симпсона

$$I(f) \approx \frac{h}{6} \sum_{i=0}^{N-1} [f(x_i) + 4f\Big(\frac{x_i + x_{i+1}}{2}\Big) + f(x_{i+1})]$$

и применим правило Рунге для сравнения полученных результатов.

Задание.

1. Вычислить значения подинтегральной функции $f_i = f(x_i)$ на равномерной сетке $x_i = a + ih$, $h = (b - a)/N$, $i = 0, 1, \ldots, N$, и найти приближенное значение интеграла $I(f)$ по составной формуле трапеций.

2. Вычислить приближенное значение интеграла $I(f)$ по составной формуле Симпсона.

3. Сравнить полученные результаты, применив правило Рунге.

Варианты.

№ 1. $\displaystyle\int_{0,8}^{1,6} \frac{dx}{\sqrt{2x^2 + 1}}$.

№ 2. $\displaystyle\int_{1,2}^{2} \frac{\lg(x + 2)}{x} dx$.

№ 3. $\displaystyle\int_{1,2}^{2,7} \frac{dx}{\sqrt{x^2 + 3{,}2}}$.

№ 4. $\displaystyle\int_{1,6}^{2,4} (x + 1) \sin(x) dx$.

№ 5. $\displaystyle\int_{1}^{2} \frac{dx}{\sqrt{2x^2 + 1{,}3}}$.

№ 6. $\displaystyle\int_{0,2}^{1} \frac{\mathrm{tg}(x^2)}{x^2 + 1} dx$.

№ 7. $\int_{0,2}^{1,2} \frac{dx}{\sqrt{x^2+1}}$.

№ 8. $\int_{0,6}^{1,4} \frac{\cos(x)}{x+1}dx$.

№ 9. $\int_{0,8}^{1,4} \frac{dx}{\sqrt{2x^2+3}}$.

№ 10. $\int_{0,4}^{1,2} \sqrt{x}\cos(x^2)dx$.

№ 11. $\int_{0,4}^{1,2} \frac{dx}{\sqrt{2+0{,}5x^2}}$.

№ 12. $\int_{0,8}^{1,2} \frac{\sin(2x)}{x^2}dx$.

№ 13. $\int_{1,4}^{2,1} \frac{dx}{\sqrt{3x^2-1}}$.

№ 14. $\int_{0,8}^{1,6} \frac{\lg(x^2+1)}{x}dx$.

№ 15. $\int_{1,2}^{2,4} \frac{dx}{\sqrt{0{,}5+x^2}}$.

№ 16. $\int_{0,4}^{1,2} \frac{\cos(x)}{x+2}dx$.

Образец.

В качестве подинтегральной функции возьмем

$$f(x) = \frac{1}{(x-0{,}3)^2+0{,}01} + \frac{1}{(x-0{,}9)^2+0{,}04} - 6, \quad 0 \le x \le 1.$$

Для приближенного вычисления интеграла $I(f)$ по составной формуле трапеций воспользуемся программой Матлаба trap(x,f), которая требует задания массивов x_i и $f(x_i)$.

Листинг M-файла «trap.m»

```
function trap;
% Составной метод трапеций для приближенного вычисления
% определенных интегралов
x=0:0.001:1;
f=((x-0.3).^2+0.01).^(-1)+((x-0.9).^2+0.04).^(-1)-6;
trap(x,f)
```

Программа trap дает приближенное значение интеграла 29,8571.

Функция Матлаба quad позволяет вычислить тот же интеграл $I(f)$ по составной формуле Симпсона. Стандартное обращение к этой программе имеет вид q=quad(@fun,a,b). Здесь требуется явное задание подинтегральной функции fun в виде M-файла.

Листинг M-файла «humps.m»

```
function f=humps(x);
f=((x-0.3).^2+0.01).^(-1)+((x-0.9).^2+0.04).^(-1)-6;
```

Чтобы проинтегрировать на отрезке $[0,1]$ функцию, определенную как humps.m, обращаемся к программе quad в виде quad(@humps,0,1). Получаем приближенное значение интеграла 29,8583. Таким образом, отличие от предыдущего результата

составляет 0,0012. Результат по формуле Симпсона также получим, применив к формуле трапеций правило Рунге.

Лабораторная работа № 6
Решение систем линейных уравнений

Постановка задачи. Найти решение системы линейных уравнений

$$\mathbf{A}\mathbf{x} = \mathbf{b},$$

и исследовать свойства матрицы $\mathbf{A}$.

Задание.

1. Решить систему $\mathbf{A}\mathbf{x} = \mathbf{b}$ методом исключения Гаусса с выбором ведущих элементов по столбцам.

2. Вычислить $\det(\mathbf{A})$ и $||\mathbf{A}||_k$ для $k = 1, 2, \infty$.

3. Найти $\mathbf{A}^{-1}$ и $\mathrm{cond}_k(\mathbf{A})$, $k = 1, 2, \infty$.

4. Выполнить LU-разложение матрицы $\mathbf{A}$.

5. Получить QR-разложение матрицы $\mathbf{A}$.

Варианты.

№ 1. $\mathbf{A} = \begin{pmatrix} 4{,}4 & -2{,}5 & 19{,}2 & -10{,}8 \\ 5{,}5 & -9{,}3 & -14{,}2 & 13{,}2 \\ 7{,}1 & -11{,}5 & 5{,}3 & -6{,}7 \\ 14{,}2 & 23{,}4 & -8{,}8 & 5{,}3 \end{pmatrix}$, $\mathbf{b} = \begin{pmatrix} 4{,}3 \\ 6{,}8 \\ -1{,}8 \\ 7{,}2 \end{pmatrix}$.

№ 2. $\mathbf{A} = \begin{pmatrix} 8{,}2 & -3{,}2 & 14{,}2 & 14{,}8 \\ 5{,}6 & -12 & 15 & -6{,}4 \\ 5{,}7 & 3{,}6 & -12{,}4 & -2{,}3 \\ 6{,}8 & 13{,}2 & -6{,}3 & -8{,}7 \end{pmatrix}$, $\mathbf{b} = \begin{pmatrix} -8{,}4 \\ 4{,}5 \\ 3{,}3 \\ 14{,}3 \end{pmatrix}$.

№ 3. $\mathbf{A} = \begin{pmatrix} 5{,}7 & -7{,}8 & -5{,}6 & -8{,}3 \\ 6{,}6 & 13{,}1 & -6{,}3 & 4{,}3 \\ 14{,}7 & -2{,}8 & 5{,}6 & -12{,}1 \\ 8{,}5 & 12{,}7 & -23{,}7 & 5{,}7 \end{pmatrix}$, $\mathbf{b} = \begin{pmatrix} 2{,}7 \\ -5{,}5 \\ 8{,}6 \\ 14{,}7 \end{pmatrix}$.

№ 4. $\mathbf{A} = \begin{pmatrix} 3{,}8 & 14{,}2 & 6{,}3 & -15{,}5 \\ 8{,}3 & -6{,}6 & 5{,}8 & 12{,}2 \\ 6{,}4 & -8{,}5 & -4{,}3 & 8{,}8 \\ 17{,}1 & -8{,}3 & 14{,}4 & -7{,}2 \end{pmatrix}$, $\mathbf{b} = \begin{pmatrix} 2{,}8 \\ -4{,}7 \\ 7{,}7 \\ 13{,}5 \end{pmatrix}$.

№ 5. $\mathbf{A} = \begin{pmatrix} 15{,}7 & 6{,}6 & -5{,}7 & 11{,}5 \\ 8{,}8 & -6{,}7 & 5{,}5 & -4{,}5 \\ 6{,}3 & -5{,}7 & -23{,}4 & 6{,}6 \\ 14{,}3 & 8{,}7 & -15{,}7 & -5{,}8 \end{pmatrix}$, $\mathbf{b} = \begin{pmatrix} -2{,}4 \\ 5{,}6 \\ 7{,}7 \\ 23{,}4 \end{pmatrix}$.

№ 6. $\mathbf{A} = \begin{pmatrix} 4{,}3 & -12{,}1 & 23{,}2 & -14{,}1 \\ 2{,}4 & -4{,}4 & 3{,}5 & 5{,}5 \\ 5{,}4 & 8{,}3 & -7{,}4 & -12{,}7 \\ 6{,}3 & -7{,}6 & 1{,}34 & 3{,}7 \end{pmatrix}$, $\mathbf{b} = \begin{pmatrix} 15{,}5 \\ 2{,}5 \\ 8{,}6 \\ 12{,}1 \end{pmatrix}$.

№ 7. $\mathbf{A} = \begin{pmatrix} 14{,}4 & -5{,}3 & 14{,}3 & -12{,}7 \\ 23{,}4 & -14{,}2 & -5{,}4 & 2{,}1 \\ 6{,}3 & -13{,}2 & -6{,}5 & 14{,}3 \\ 5{,}6 & 8{,}8 & -6{,}7 & -23{,}8 \end{pmatrix}$, $\mathbf{b} = \begin{pmatrix} -14{,}4 \\ 6{,}6 \\ 9{,}4 \\ 7{,}3 \end{pmatrix}$.

№ 8. $\mathbf{A} = \begin{pmatrix} 1{,}7 & 10 & -1{,}3 & 2{,}1 \\ 3{,}1 & 1{,}7 & -2{,}1 & 5{,}4 \\ 3{,}3 & -7{,}7 & 4{,}4 & -5{,}1 \\ 10 & -20{,}1 & 20{,}4 & 1{,}7 \end{pmatrix}$, $\mathbf{b} = \begin{pmatrix} 3{,}1 \\ 2{,}1 \\ 1{,}9 \\ 1{,}8 \end{pmatrix}$.

№ 9. $\mathbf{A} = \begin{pmatrix} 1{,}7 & -1{,}8 & 1{,}9 & -57{,}4 \\ 1{,}1 & -4{,}3 & 1{,}5 & -1{,}7 \\ 1{,}2 & 1{,}4 & 1{,}6 & 1{,}8 \\ 7{,}1 & -1{,}3 & -4{,}1 & 5{,}2 \end{pmatrix}$, $\mathbf{b} = \begin{pmatrix} 10 \\ 19 \\ 20 \\ 10 \end{pmatrix}$.

№ 10. $\mathbf{A} = \begin{pmatrix} 6{,}1 & 6{,}2 & -6{,}3 & 6{,}4 \\ 1{,}1 & -1{,}5 & 2{,}2 & -3{,}8 \\ 5{,}1 & -5{,}0 & 4{,}9 & -4{,}8 \\ 1{,}8 & 1{,}9 & 2{,}0 & -2{,}1 \end{pmatrix}$, $\mathbf{b} = \begin{pmatrix} 6{,}5 \\ 4{,}2 \\ 4{,}7 \\ 2{,}2 \end{pmatrix}$.

№ 11. $\mathbf{A} = \begin{pmatrix} 2{,}2 & -3{,}1 & 4{,}2 & -5{,}1 \\ 1{,}3 & 2{,}2 & -1{,}4 & 1{,}5 \\ 6{,}2 & -7{,}4 & 8{,}5 & -9{,}6 \\ 1{,}2 & 1{,}3 & 1{,}4 & 4{,}5 \end{pmatrix}$, $\mathbf{b} = \begin{pmatrix} 6{,}01 \\ 10 \\ 1{,}1 \\ 1{,}6 \end{pmatrix}$.

№ 12. $\mathbf{A} = \begin{pmatrix} 35{,}8 & 2{,}1 & -34{,}5 & -11{,}8 \\ 27{,}1 & -7{,}5 & 11{,}7 & -23{,}5 \\ 11{,}7 & 1{,}8 & -6{,}5 & 7{,}1 \\ 6{,}3 & 10 & 7{,}1 & 3{,}4 \end{pmatrix}$, $\mathbf{b} = \begin{pmatrix} 0{,}5 \\ 12{,}8 \\ 1{,}7 \\ 20{,}8 \end{pmatrix}$.

№ 13. $\mathbf{A} = \begin{pmatrix} 35{,}1 & 1{,}7 & 37{,}5 & -2{,}8 \\ 45{,}2 & 21{,}1 & -1{,}1 & -1{,}2 \\ -21{,}1 & 31{,}7 & 1{,}2 & -1{,}5 \\ 31{,}7 & 18{,}1 & -31{,}7 & 2{,}2 \end{pmatrix}$, $\mathbf{b} = \begin{pmatrix} 7{,}5 \\ 11{,}1 \\ 2{,}1 \\ 0{,}5 \end{pmatrix}$.

№ 14. $\mathbf{A} = \begin{pmatrix} 1{,}1 & 11{,}2 & 11{,}1 & -13{,}1 \\ -3{,}3 & 1{,}1 & 30{,}1 & -20{,}1 \\ 7{,}5 & 1{,}3 & 1{,}1 & 10 \\ 1{,}7 & 7{,}5 & -1{,}8 & 2{,}1 \end{pmatrix}$, $\mathbf{b} = \begin{pmatrix} 1{,}3 \\ 1{,}1 \\ 20 \\ 1{,}1 \end{pmatrix}$.

№ 15. $\mathbf{A} = \begin{pmatrix} 7{,}5 & 1{,}8 & -2{,}1 & -7{,}7 \\ -10 & 1{,}3 & -20 & -1{,}4 \\ 2{,}8 & -1{,}7 & 3{,}9 & 4{,}8 \\ 10 & 31{,}4 & -2{,}1 & -10 \end{pmatrix}$, $\mathbf{b} = \begin{pmatrix} 1{,}1 \\ 1{,}5 \\ 1{,}2 \\ -1{,}1 \end{pmatrix}$.

№ 16. $\mathbf{A} = \begin{pmatrix} 30{,}1 & -1{,}4 & 10 & -1{,}5 \\ -17{,}5 & 11{,}1 & 1{,}3 & -7{,}5 \\ 1{,}7 & -21{,}1 & 7{,}1 & -17{,}1 \\ 2{,}1 & 2{,}1 & 3{,}5 & 3{,}3 \end{pmatrix}$, $\mathbf{b} = \begin{pmatrix} 10 \\ 1{,}3 \\ 10 \\ 1{,}7 \end{pmatrix}$.

Образец.

Рассмотрим линейную систему $\mathbf{Ax} = \mathbf{b}$, где

$$\mathbf{A} = \begin{pmatrix} 14 & -8 & -21 & 12 \\ 10 & -6 & -15 & 9 \\ 35 & -20 & -56 & 32 \\ 25 & -15 & -40 & 24 \end{pmatrix}, \qquad \mathbf{b} = \begin{pmatrix} 19 \\ 14 \\ 53 \\ 39 \end{pmatrix}.$$

1. Решение этой системы по методу Гаусса с выбором ведущих элементов по столбцам в Матлабе осуществляется с помощью команды x = A\b. В нашем случае это дает решение $\mathbf{x} = (-97; -160; -61; -99)^T$.

2. Используя команду det(A), получаем $\det(\mathbf{A}) = 1$. Аналогично находим основные нормы матрицы $\mathbf{A}$: norm(A,1)=132, norm(A,2)=100,4788, norm(A,inf)=143.

3. Применяя команду inv(A), получаем $\mathbf{A}^{-1} = \begin{pmatrix} 24 & -32 & -9 & 12 \\ 40 & -56 & -15 & 21 \\ 15 & -20 & -6 & 8 \\ 25 & -35 & -10 & 14 \end{pmatrix}$. Для чисел обусловленности матрицы $\mathbf{A}$ имеем cond(A,2)=1,0096 $\cdot$ 10^4, cond(A,1)= =cond(A,inf)=1,8870 $\cdot$ 10^4.

4. LU-разложение матрицы $\mathbf{A}$ получаем по команде [L,U]=lu(A):

$$\mathbf{L} = \begin{pmatrix} 0{,}4000 & 0{,}0000 & 1{,}0000 & 0 \\ 0{,}2857 & 0{,}4000 & 0{,}7143 & 1{,}0000 \\ 1{,}0000 & 0 & 0 & 0 \\ 0{,}7143 & 1{,}0000 & 0 & 0 \end{pmatrix},$$

$$\mathbf{U} = \begin{pmatrix} 35{,}0000 & -20{,}0000 & -56{,}0000 & 32{,}0000 \\ 0 & -0{,}7143 & 9 & 1{,}1429 \\ 0 & 0 & 1{,}4000 & -0{,}8000 \\ 0 & 0 & 0 & -0{,}0286 \end{pmatrix}.$$

Здесь матрица $\mathbf{L}$ не является нижней треугольной, так как при работе алгоритма исключения Гаусса были использованы перестановки строк. Для получения нижней треугольной матрицы $\mathbf{L}$ пользуемся командой [L, U, P]=lu(A); L, P. При этом

матрица $\mathbf{U}$ естественно остается той же самой и поэтому повторно не печатается. Так как имеет место равенство $\mathbf{LU} = \mathbf{PA}$, где $\mathbf{P}$ – матрица перестановок строк, то получаем

$$\mathbf{L} = \begin{pmatrix} 1{,}0000 & 0 & 0 & 0 \\ 0{,}7143 & 1{,}0000 & 0 & 0 \\ 0{,}4000 & 0 & 1{,}0000 & 0 \\ 0{,}2857 & 0{,}4000 & 0{,}7143 & 1{,}0000 \end{pmatrix}, \quad \mathbf{P} = \begin{pmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{pmatrix}.$$

5. QR-разложение матрицы $\mathbf{A}$ находим по команде [Q,R]=qr(A):

$$\mathbf{Q} = \begin{pmatrix} -0{,}3022 & 0{,}2159 & 0{,}7555 & 0{,}5397 \\ -0{,}2159 & -0{,}3022 & 0{,}5397 & -0{,}7555 \\ -0{,}7555 & 0{,}5397 & -0{,}3022 & -0{,}2159 \\ -0{,}5397 & -0{,}7555 & -0{,}2159 & 0{,}3022 \end{pmatrix},$$

$$\mathbf{R} = \begin{pmatrix} -46{,}3249 & 26{,}9185 & 73{,}4809 & -42{,}6984 \\ 0 & 0{,}6260 & -0{,}0000 & -0{,}9930 \\ 0 & 0 & 1{,}5974 & -0{,}9282 \\ 0 & 0 & 0 & 0{,}0216 \end{pmatrix}.$$

Лабораторная работа № 7

Решение задач на собственные значения

Постановка задачи. Найти собственные значения и собственные векторы вещественной симметрической матрицы $\mathbf{A}$.

Задание.

1. Найти наибольшее по модулю собственное значение и соответствующий ему собственный вектор степенным методом. В качестве начального приближения использовать вектор $(1; 1; 1; 1)^T$.

2. Найти наименьшее по модулю собственное значение и соответствующий ему собственный вектор обратным степенным методом.

3. Решить полную проблему собственных значений методом вращений Якоби. Точность вычислений в евклидовой норме $\varepsilon = 10^{-6}$.

4. Получить сингулярное разложение матрицы $\mathbf{A}$.

Варианты.

№ 1. $\mathbf{A} = \begin{pmatrix} 1 & 1{,}5 & 2{,}5 & 3{,}5 \\ 1{,}5 & 1 & 2 & 1{,}6 \\ 2{,}5 & 2 & 1 & 1{,}7 \\ 3{,}5 & 1{,}6 & 1{,}7 & 1 \end{pmatrix}$. № 2. $\mathbf{A} = \begin{pmatrix} 1 & 1{,}2 & 2 & 0{,}5 \\ 1{,}2 & 1 & 0{,}4 & 1{,}2 \\ 2 & 0{,}4 & 2 & 1{,}5 \\ 0{,}5 & 1{,}2 & 1{,}5 & 1 \end{pmatrix}$.

№ 3. $\mathbf{A} = \begin{pmatrix} 1 & 1{,}2 & 2 & 0{,}5 \\ 1{,}2 & 1 & 0{,}5 & 1 \\ 2 & 0{,}5 & 2 & 1{,}5 \\ 0{,}5 & 1 & 1{,}5 & 0{,}5 \end{pmatrix}$.

№ 4. $\mathbf{A} = \begin{pmatrix} 2{,}5 & 1 & -0{,}5 & 2 \\ 1 & 2 & 1{,}2 & 0{,}4 \\ -0{,}5 & 1{,}2 & -1 & 1{,}5 \\ 2 & 0{,}4 & 1{,}5 & 1 \end{pmatrix}$.

№ 5. $\mathbf{A} = \begin{pmatrix} 2 & 1 & 1{,}4 & 0{,}5 \\ 1 & 1 & 0{,}5 & 1 \\ 1{,}4 & 0{,}5 & 2 & 1{,}2 \\ 0{,}5 & 1 & 1{,}2 & 0{,}5 \end{pmatrix}$.

№ 6. $\mathbf{A} = \begin{pmatrix} 2 & 1{,}2 & -1 & 1 \\ 1{,}2 & 0{,}5 & 2 & -1 \\ -1 & 2 & -1{,}5 & 0{,}2 \\ 1 & -1 & 0{,}2 & 1{,}5 \end{pmatrix}$.

№ 7. $\mathbf{A} = \begin{pmatrix} 2 & 1{,}5 & 3{,}5 & 4{,}5 \\ 1{,}5 & 2 & 2 & 1{,}6 \\ 3{,}5 & 2 & 2 & 1{,}7 \\ 4{,}5 & 1{,}6 & 1{,}7 & 2 \end{pmatrix}$.

№ 8. $\mathbf{A} = \begin{pmatrix} 1 & 0{,}5 & 1{,}2 & -1 \\ 0{,}5 & 2 & -0{,}5 & 0 \\ 1{,}2 & -0{,}5 & -1 & 1{,}4 \\ -1 & 0 & 1{,}4 & 1 \end{pmatrix}$.

№ 9. $\mathbf{A} = \begin{pmatrix} 1{,}2 & 0{,}5 & 2 & 1 \\ 0{,}5 & 1 & 0{,}8 & 2 \\ 2 & 0{,}8 & 1 & 1 \\ 1 & 2 & 1 & 2 \end{pmatrix}$.

№ 10. $\mathbf{A} = \begin{pmatrix} 0{,}5 & 1{,}2 & 1 & 0{,}9 \\ 1{,}2 & 2 & 0{,}5 & 1{,}2 \\ 1 & 0{,}5 & 1 & 1 \\ 0{,}5 & 1{,}2 & 1 & 2{,}2 \end{pmatrix}$.

№ 11. $\mathbf{A} = \begin{pmatrix} 1{,}2 & 0{,}5 & 2 & 1 \\ 0,5 & 1 & 0{,}6 & 2 \\ 2 & 0{,}6 & 1 & 1 \\ 1 & 2 & 1 & 1{,}3 \end{pmatrix}$.

№ 12. $\mathbf{A} = \begin{pmatrix} 2 & 1{,}5 & 4{,}5 & 5{,}5 \\ 1{,}5 & 3 & 2 & 1{,}6 \\ 4{,}5 & 2 & 3 & 1{,}7 \\ 5{,}5 & 1{,}6 & 1{,}7 & 3 \end{pmatrix}$.

№ 13. $\mathbf{A} = \begin{pmatrix} 1{,}6 & 1 & 1{,}4 & 1 \\ 1 & 1 & 0{,}5 & 2 \\ 1{,}4 & 0{,}5 & 2 & 1{,}2 \\ 1 & 2 & 1{,}2 & 0{,}5 \end{pmatrix}$.

№ 14. $\mathbf{A} = \begin{pmatrix} 2{,}4 & 0{,}5 & 2 & 1 \\ 0{,}5 & 1 & 0{,}8 & 2 \\ 2 & 0{,}8 & 1 & 0{,}5 \\ 1 & 2 & 0{,}5 & 1{,}2 \end{pmatrix}$.

№ 15. $\mathbf{A} = \begin{pmatrix} 0{,}5 & 1{,}2 & 2 & 1 \\ 1{,}2 & 2 & 0{,}5 & 1{,}2 \\ 2 & 0{,}5 & 1 & 0{,}5 \\ 1 & 1{,}2 & 0{,}5 & 1{,}6 \end{pmatrix}$.

№ 16. $\mathbf{A} = \begin{pmatrix} 1{,}8 & 1{,}6 & 1{,}7 & 1{,}8 \\ 1{,}6 & 2{,}8 & 1{,}5 & 1{,}3 \\ 1{,}7 & 1{,}5 & 3{,}8 & 1{,}4 \\ 1{,}8 & 1{,}3 & 1{,}4 & 4{,}8 \end{pmatrix}$.

Указания.

Рассмотрим матрицу

$$\mathbf{A} = \begin{pmatrix} 4 & -30 & 60 & -35 \\ -30 & 300 & -675 & 420 \\ 60 & -675 & 1620 & -1050 \\ -35 & 420 & -1050 & 700 \end{pmatrix}.$$

Нахождение собственных значений и собственных векторов матрицы $\mathbf{A}$ в Матлабе осуществляется с помощью команды [V,D]=eig(A). В нашем случае это дает решение

$$\mathbf{V} = \begin{pmatrix} 0{,}7926 & 0{,}5821 & 0{,}1792 & -0{,}0292 \\ 0{,}4519 & -0{,}3705 & -0{,}7419 & 0{,}3287 \\ 0{,}3224 & -0{,}5036 & 0{,}1002 & -0{,}7914 \\ 0{,}2522 & -0{,}5140 & 0{,}6383 & 0{,}5146 \end{pmatrix},$$

$$\mathbf{D} = 10^3 \cdot \begin{pmatrix} 0{,}0002 & 0 & 0 & 0 \\ 0 & 0{,}0015 & 0 & 0 \\ 0 & 0 & 0{,}0371 & 0 \\ 0 & 0 & 0 & 2{,}5853 \end{pmatrix}.$$

Если требуется найти только собственные значения, то можно воспользоваться командой lambda=eig(A), что дает

$$\lambda = 10^3 \cdot (0{,}0002; 0{,}0015; 0{,}0371; 2{,}5853).$$

После 19 итераций по методу вращений Якоби получаем:

$$\begin{aligned} &\lambda_1 = 2585{,}2538, && \mathbf{e}_1 = (0{,}0292; -0{,}3287; 0{,}7914; -0{,}5146)^T, \\ &\lambda_2 = 37{,}1015, && \mathbf{e}_2 = (-0{,}1792; 0{,}7419; -0{,}1002; -0{,}6383)^T, \\ &\lambda_3 = 1{,}4781, && \mathbf{e}_3 = (-0{,}5821; 0{,}3705; 0{,}5096; 0{,}5140)^T, \\ &\lambda_4 = 0{,}1666, && \mathbf{e}_4 = (0{,}7926; 0{,}4519; 0{,}3224; 0{,}2522)^T. \end{aligned}$$

Используя команду Матлаба [U,S,V]=svd(A), находим сингулярное разложение матрицы $\mathbf{A} = \mathbf{USV}$, где

$$\mathbf{U} = \mathbf{V} = \begin{pmatrix} -0{,}0292 & 0{,}1792 & 0{,}5821 & 0{,}7926 \\ 0{,}3287 & -0{,}7419 & -0{,}3705 & 0{,}4519 \\ -0{,}7914 & 0{,}1002 & -0{,}5096 & 0{,}3224 \\ 0{,}5146 & 0{,}6383 & -0{,}5140 & 0{,}2522 \end{pmatrix},$$

$$\mathbf{S} = 10^3 \cdot \begin{pmatrix} 2{,}5853 & 0 & 0 & 0 \\ 0 & 0{,}0371 & 0 & 0 \\ 0 & 0 & 0{,}0015 & 0 \\ 0 & 0 & 0 & 0{,}0002 \end{pmatrix}.$$

Приложение B
Тесты для письменного экзамена

1. Найти значение переменной x после выполнения следующего псевдокода?

```
x=0;
for  i= 1:5
     if  i <= 3  x=x+1
     else  x=2*x end
```

(1) 3 ; (2) 5 ; (3) 7 ; (4) 12 ; (5) 16.

2. Рассмотреть уравнение

$$x^2 = e^{-x}.$$

Используя график, определить число корней этого уравнения.

(1) корней нет;

(2) один простой ;

(3) один кратный корень;

(4) один простой корень и один кратный корень;

(5) два простых корня.

3. Уравнение $1 - 3x + \sin x = 0$ имеет корень на отрезке $[0, 1]$. Сколько итераций метода проб требуется, чтобы получить этот корень с точностью $\frac{1}{2}10^{-5}$?

(1) 3 ; (2) 5 ; (3) 8 ; (4) 17 ; (5) 24.

4. Найти минимальное число итераций метода проб, требуемое для уменьшения длины отрезка, содержащего корень, до $1/50$ его исходной длины?

(1) 5 ; (2) 6 ; (3) 12 ; (4) 25 ; (5) 50.

5. Укажите правильную формулу для метода хорд (ложного положения).

(1) $c_n = b_n - \dfrac{f(a_n)(b_n - a_n)}{f(b_n) - f(a_n)}$;

(2) $c_n = b_n - \dfrac{f(b_n)(b_n - a_n)}{f(b_n) - f(a_n))}$;

(3) $c_n = b_n - \dfrac{f(b_n) - f(a_n)}{f(b_n)(b_n - a_n)}$;

(4) $c_n = a_n - \dfrac{f(b_n)(f(b_n) - f(a_n))}{b_n - a_n}$;

(5) $c_n = a_n - \dfrac{f(b_n) - f(a_n)}{f(b_n)(b_n - a_n)}$.

6. Метод хорд, примененный к уравнению $f(x) = 2x^2 - 1 = 0$ на отрезке $[0, 1]$, после двух итераций дает приближенное значение корня:

(1) 0 ; (2) $\frac{1}{8}$; (3) $\frac{1}{4}$; (4) $\frac{1}{2}$; (5) $\frac{2}{3}$.

7. Применение метода Ньютона для вычисления значения $\sqrt{2}$ дает итерационную формулу:

$$(1)\ x_{i+1} = \frac{x_i^2 + 2}{2x_i}; \quad (2)\ x_{i+1} = \frac{3x_i^2 - 2}{2x_i}; \quad (3)\ x_{i+1} = x_i - \sqrt{2}x_i;$$

$$(4)\ x_{i+1} = \frac{x_i^3 + 2\sqrt{2}}{3x_i^2}; \quad (5)\ \text{все неверны.}$$

8. Указать правильную формулу для метода секущих:

(1) $x_{n+1} = x_{n-1} - \dfrac{f(x_n)(x_n - x_{n-1})}{f(x_n) - f(x_{n-1})}$;

(2) $x_{n+1} = x_n - \dfrac{f(x_n)(x_n - x_{n-1})}{f(x_n) - f(x_{n-1})}$;

(3) $x_{n+1} = x_n - \dfrac{f(x_{n-1})(x_n - x_{n-1})}{f(x_n) - f(x_{n-1})}$;

(4) $x_{n+1} = x_n - \dfrac{f(x_n) - f(x_{n-1})}{(x_n - x_{n-1})f(x_{n-1})}$;

(5) $x_{n+1} = x_{n-1} - \dfrac{f(x_{n-1})(x_n - x_{n-1})}{f(x_n) - f(x_{n-1})}$.

9. Рассмотреть применение метода секущих для решения уравнения вида $x^3 - 3x + 2 = 0$. Используя $x_0 = 0$, $x_1 = \frac{1}{2}$, после двух итераций получаем:

(1) $\frac{5}{9}$; (2) $\frac{8}{11}$; (3) $\frac{3}{4}$; (4) $\frac{4}{5}$; (5) 1.

10. Что вы думаете о сходимости (расходимости) метода простой итерации $x_{n+1} = g(x_n)$, $n = 0, 1, \ldots$, где $g(x) = e^{-x}$ и $x_0 = 1$.

(a) монотонная сходимость;

(b) монотонная расходимость;

(c) осциллирующая сходимость;

(d) осциллирующая расходимость;

(e) поведение неопределено.

11. Метод простой итерации применяется для решения уравнения $x(x - 2) = 3$. Используя уравнение $x = g(x) = 2 + \frac{3}{x}$ и $x_0 = 1$, после двух итераций получаем:

(1) 5 ; (2) $3\frac{1}{5}$; (3) 3 ; (4) $2\frac{3}{5}$; (5) $2\frac{6}{7}$.

12. Каково минимальное число арифметических операций, нужное для вычисления значения многочлена

$$p(x) = x^5 - 5x^4 + 5x^3 + 5x^2 - 6x + 1$$

при $x = \alpha$.

(a) 6; (b) 9; (c) 12; (d) 15; (e) 19.

13. Методом исключения Гаусса решить систему линейных уравнений (a – постоянная)

$$\begin{aligned} x_1 + 6x_2 - 2x_3 &= 5a \\ 2x_1 + x_2 - 2x_3 &= 1 \\ 2x_1 + 2x_2 + 6x_3 &= 10. \end{aligned}$$

Указать правильное решение этой системы:

(1) $(1, 1, a)^T$;

(2) $\frac{1}{9}(14 - 5a, 1 + 8a, 11 - a)^T$;

(3) $\frac{1}{9}(10a - 1, 2a + 7, 11a - 2)^T$;

(4) $\frac{1}{9}(10a + 1, 2a - 7, 11a + 2)^T$;

(5) все неверны.

14. Каково общее число арифметических операций в методе исключения Гаусса?

(1) $\frac{1}{6}(4n^3 + 3n^2 - 7n)$;

(2) n^2 ;

(3) $\frac{1}{6}(4n^3 + 9n^2 - 7n)$;

(4) $\frac{2}{3}n^3$;

(5) $\frac{1}{6}(5n^3 - 3n^2 - 7n)$.

15. Рассмотреть матричное уравнение

$$\begin{pmatrix} 10 & 2 & 11 \\ 10 & 1 & 1 \\ -1 & 10 & 1 \end{pmatrix} \begin{pmatrix} x \\ y \\ z \end{pmatrix} = \begin{pmatrix} -1 \\ 9 \\ -1 \end{pmatrix}.$$

Какое из приводимых утверждений является верным?

(1) Уравнения можно переупорядочить таким образом, что матрица будет иметь диагональное преобладание.

(2) Диагональное преобладание можно получить, производя вычитание одной строки и перестановку других строк.

(3) Решение нельзя получить без использования выбора ведущего элемента.

(4) Решение имеет вид $(1, 0, -1)^T$.

(5) Для получения решения методом Гаусса требуется < 28 операций.

16. Рассмотреть матричное уравнение

$$\begin{pmatrix} 4 & 1 & 0 & 0 \\ 1 & 4 & 1 & 0 \\ 0 & 1 & 4 & 1 \\ 0 & 0 & 1 & 4 \end{pmatrix} \begin{pmatrix} x \\ y \\ z \\ w \end{pmatrix} = \begin{pmatrix} 5 \\ 6 \\ 6 \\ 5 \end{pmatrix}.$$

Сколько арифметических операций требуется для решения этой трехдиагональной системы методом прогонки?

(1) 18 ; (2) 25 ; (3) 36 ; (4) 62 ; (5) 90.

17. Каково число обусловленности $cond(A)$ матрицы

$$A = \begin{pmatrix} 2 & 1 \\ 1 & 2 \end{pmatrix}?$$

(1) 1 ; (2) 2 ; (3) 3 ; (4) 4 ; (5) 10.

18. Две итерации метода Якоби для нахождения приближенного решения системы уравнений

$$\begin{pmatrix} 2 & 1 \\ 1 & 3 \end{pmatrix} \begin{pmatrix} x \\ y \end{pmatrix} = \begin{pmatrix} 5 \\ 5 \end{pmatrix}$$

с начальными значениями $x^{(0)} = y^{(0)} = 0$ дают:

(1) $x^{(2)} = 5/2$, $y^{(2)} = 5/3$;

(2) $x^{(2)} = 5/3$, $y^{(2)} = 5/6$;

(3) $x^{(2)} = 5/6$, $y^{(2)} = 5/3$;

(4) $x^{(2)} = 5/3$, $y^{(2)} = 5/2$;

(5) все неверны.

19. Какая из приводимых матриц подходит при использовании метода Зейделя для нахождения приближенного решения линейной системы $Ax = b$?

(1) $\begin{pmatrix} 1 & 1 \\ 2 & 0 \end{pmatrix}$; (2) $\begin{pmatrix} 1 & 2 \\ 2 & 1 \end{pmatrix}$; (3) $\begin{pmatrix} 3 & -1 \\ 1 & -2 \end{pmatrix}$; (4) $\begin{pmatrix} 1 & 4 \\ 1 & 4 \end{pmatrix}$; (5) все не подходят.

Список литературы

[1] Бахвалов Н. С. и др. Численные методы / Н. С. Бахвалов, Н. П. Жидков, Г. М. Кобельков. М.: БИНОМ. Лаборатория знаний, 2007.

[2] Бахвалов Н. С. и др. Численные методы в задачах и упражнениях / Н. С. Бахвалов, А. В. Лапин, Е. В. Чижонков. М.: Высшая школа, 2000.

[3] Березин И. С., Жидков Н. П. Методы вычислений, т. 1, М.: Наука, 1966; т. 2, М.: Физматлит, 1962.

[4] Вазов В., Форсайт Дж. Разностные методы решения дифференциальных уравнений в частных производных. М.: ИЛ, 1963.

[5] Вержбицкий В. М. Численные методы. Линейная алгебра и нелинейные уравнения. М.: ОНИКС 21 век, 2005.

[6] Воеводин В. В., Кузнецов Ю. А. Матрицы и вычисления. М.: Наука, 1984.

[7] Воробьева Г. Н., Данилова А. Н. Практикум по вычислительной математике. М.: Высшая школа, 1990.

[8] Голуб Дж., Ван Лоун Ч. Матричные вычисления. М.: Мир, 1999.

[9] Годунов С. К. Современные аспекты линейной алгебы. Новосибирск: Научная книга, 1997.

[10] Деммель Дж. Вычислительная линейная алгебра. М.: Мир, 2001.

[11] Джордж А., Лю Дж. Численное решение больших разреженных систем уравнений. М.: Мир, 1984.

[12] Дробышевич В. И. и др. Задачи по вычислительной математике / В. И. Дробышевич, В. П. Дымников, Г. С. Ривин. М.: Наука, 1980.

[13] Завьялов Ю. С. и др. Методы сплайн-функций / Ю. С. Завьялов, Б. И. Квасов, В. Л. Мирошниченко. М.: Наука, 1980.

[14] Калиткин Н. Н. Численные методы. М.: Наука, 1978.

[15] Каханер Д. и др. Численные методы и программное обеспечение / Д. Каханер, Л. Моулер, С. М. Нэш. М.: Мир, 2001.

[16] Коновалов А. Н. Введение в вычислительные методы линейной алгебры. Новосибирск: ВО Наука, 1993.

[17] Квасов Б. И. Методы изогеометрической аппроксимации сплайнами. М.: Физматлит, 2006.

[18] Крылов В. И. и др. Вычислительные методы / В. И. Крылов, В. В. Бобков, П. И. Монастырный. М.: Наука, 1976.

[19] Мак-Кракен Д., Дорн У. Численные методы и программирование на Фортране. М.: Мир, 1977.

[20] Миньков С. Л., Миньков Л. Л. Основы численных методов. Томск: ТГУ, 2006.

[21] Мэтьюз Дж. Г., Финк К. Д. Численные методы. Использование MATLAB. 3-е издание. М.: Вильямс, 2001.

[22] Парлетт Б. Симметричная проблема собственных значений. Численные методы. М.: Мир, 1983.

[23] Самарский А. А. и др. Задачи и упражнения по численным методам / А. А. Самарский, П. Н. Вабищевич, Е. А. Самарская. М.: Едиториал УРСС, 2003.

[24] Самарский А. А., Николаев Е. С. Методы решения сеточных уравнений. М.: Наука, 1978.

[25] Стренг Г. Линейная алгебра и ее применения. М.: Мир, 1980.

[26] Тихонов А. Н., Арсенин В. Я. Методы решения некорректных задач. М.: Наука, 1979.

[27] Тыртышников Е. Е. Методы численного анализа. М.: Издательский центр «Академия», 2007.

[28] Уилкинсон Дж. Х. Алгебраическая проблема собственных значений. М.: Наука, 1970.

[29] Фаддеев Д. К., Фаддеева В. Н. Вычислительные методы линейной алгебры. М.: Физматлит, 1963.

[30] Фихтенгольц Г. М. Курс дифференциального и интегрального исчисления. Т. 1. М.: Наука, 1969.

[31] Форсайт Дж., Молер К. Численное решение систем линейных алгебраических уравнений. М.: Мир, 1969.

[32] Asaithambi N. S. Numerical analysis. Theory and practice. Fort Worth: Saunders College Publishing, 1995.

[33] Higham D. J., Higham N. J. MATLAB guide. Philadelphia: SIAM, 2000.

[34] Saad Y. Iterative Methods for Sparse Linear Systems. Boston, MA: PWS Publishing Company, 1996.

[35] Trefethen L., Bau D. Numerical linear algebra. Philadelphia, PA: SIAM, 1997.