Министерство науки и высшего образования Российской Федерации
Федеральное государственное автономное
образовательное учреждение высшего образования
«Московский физико-технический институт
(национальный исследовательский университет)»

М. Н. Петров, А. В. Чикиткин

ТРИДЦАТЬ ТРИ ЗАДАЧИ ПО ВЫЧИСЛИТЕЛЬНОЙ МАТЕМАТИКЕ

Учебное пособие

МОСКВА
МФТИ
2021

УДК 517(075)
ББК 22.19я73
П30

Рецензенты:

Кандидат физико-математических наук *А. А. Токарев*

Доктор физико-математических наук, профессор, заведующий кафедрой прикладной математики ФГБОУ ВО «МГТУ «СТАНКИН» *Л. А. Уварова*

Петров, Михаил Николаевич,
Чикиткин, Александр Викторович

П30 Тридцать три задачи по вычислительной математике : учебное пособие / М. Н. Петров, А. В. Чикиткин. – Москва : МФТИ, 2021. – 60 с.

ISBN 978-5-7417-0782-1

Подробно рассмотрены 33 задачи по вычислительной математике. Учебное пособие включает необходимый теоретический материал, позволяющий быстро разобраться в рассматриваемых задачах.

Предназначается для студентов, преподавателей вузов и всех заинтересованных в освоении вычислительной математики и математического моделирования.

Учебное издание

Петров Михаил Николаевич,
Чикиткин Александр Викторович

ТРИДЦАТЬ ТРИ ЗАДАЧИ ПО ВЫЧИСЛИТЕЛЬНОЙ МАТЕМАТИКЕ

Учебное пособие

Редактор *Н. Е. Кобзева*. Корректор *И. А. Волкова*
Компьютерная верстка *Н. Е. Кобзева*
Дизайн обложки *Е. А. Казённова*
Подписано в печать 00.09.2021. Формат $60\times84\ {}^{1}/_{16}$.
Усл. печ. л. 3,75. Уч.-изд. л. 2,9. Тираж 000 экз. Заказ № 00.

Федеральное государственное автономное образовательное учреждение высшего образования «Московский физико-технический институт (национальный исследовательский университет)»
141700, Московская обл., г. Долгопрудный, Институтский пер., 9
Тел. (495) 408-58-22, e-mail: rio@mipt.ru

Отдел оперативной полиграфии «Физтех-полиграф»
141700, Московская обл., г. Долгопрудный, Институтский пер., 9
Тел. (495) 408-84-30, e-mail: polygraph@mipt.ru

ISBN 978-5-7417-0782-1

Оглавление

ВВЕДЕНИЕ

Учебное пособие представляет собой конспект семинаров по курсу Вычислительная математика, читаемых авторами в МФТИ в осеннем семестре.

Цель пособия – познакомить читателя с минимальным набором теоретических знаний по курсу, необходимых при решении задач. Чтобы не усложнять понимание, авторы часто умышленно жертвуют математической строгостью излагаемого материала.

Данное пособие может служить отправной точкой в освоении тем курса вычислительной математики, позволяет систематизировать знания, полученные при прослушивании лекций и прочтении более канонических, математически строгих учебников.

1. Типы представления чисел

Любое вещественное число может быть представлено в виде $x = a \cdot 10^b$, где a – мантисса, b – экспонента. Например, для числа 123.456 $a = 0.123456, b = 3$. Согласно стандарту IEEE-754, вещественные числа представляются в виде $x = a \cdot 2^b$ и при хранении числа двойной точности в памяти используется 8 байт, из которых под экспоненту отводится 11 бит (1 бит под знак), а под мантиссу 53 бита (1 бит под знак), реальная длина мантиссы без знака $p = 53$, т.к. первый бит не хранится и считается равным единице. Соответственно, такой формат чисел двойной точности позволяет представлять числа в диапазоне примерно $[10^{-308}, 10^{308}]$, и верхняя граница относительной ошибки округления ϵ составляет $2^{-p} \approx 10^{-16}$.

2. Ошибки

Можно выделить следующие типы ошибок, возникающие при решении прикладной задачи на компьютере:

1. ошибка модели;
2. ошибка метода;
3. ошибка входных данных;
4. ошибка округления.

2.1. Ошибка модели

Любая математическая модель является в той или иной степени приближением реального явления или процесса. В результате этого приближения происходит потеря в точности при решении задачи уже на уровне постановки самой модели.

2.2. Ошибка метода

Поясним, что такое ошибка метода на примере вычисления производной функции в точке. Пусть необходимо вычислить значение производной функции f в точке x_0. Пусть также известно, что $f(x_0) = f_0$, а в точке $x_1 = x_0 + h$ значение функции $f(x_1) = f_1$. Предложим естественный способ вычисление производной по следующей формуле:

$$f'(x_0) \approx \frac{f_1 - f_0}{h}. \tag{1}$$

Поскольку в пределе при $h \to 0$ формула (1) и есть определение производной, то можно предположить, что при достаточно малом h по ней можно получить правдоподобное значение производной. Ошибка, которая будет возникать при вычислении производной по этой формуле и будет называться ошибкой метода.

2.3. Ошибка входных данных

Предположим, что значения функции f_0 и f_1 были получены неточно (например, в результате измерений физического процесса). Тогда, если считать производную по формуле (1) с этими значениями, то полная ошибка вычисления производной также будет включать в себя и ошибку входных данных, обусловленную неточностью задания значений функции.

2.4. Ошибка округления

При вычислении производной на компьютере также неизбежно будет возникать ошибка округления, связанная с ограничением на представление в памяти вещественного числа. В памяти компьютера два числа f и $\tilde{f} = f(1+\epsilon)$, где ϵ – машинное эпсилон, будут эквивалентны. Пусть, в результате округления в памяти, значение функции f_1 стало $f_1(1+\epsilon_1)$, а f_2 стало $f_2(1+\epsilon_2)$, где ϵ_1 и ϵ_2 по модулю ограничены машинным эпсилон. Тогда, вычисляя производную по формуле (1), получим:

$$f'(x_0) = \frac{f_1(1+\epsilon_1) - f_0(1+\epsilon_2)}{h} = \frac{f_1 - f_0}{h} + \frac{f_1\epsilon_1 - f_0\epsilon_2}{h}. \qquad (2)$$

Отсюда можно дать оценку ошибки округления

$$\epsilon_{trunc} \leq \frac{2\max(f_0, f_1)\epsilon}{h}. \qquad (3)$$

Также стоит отметить, что полная ошибка вычисления производной будет включать в себя еще и ошибку метода, обусловленную способом вычисления производной, и ошибку входных данных, если значения функции будут заданы неточно. Наличие ошибки округления показывает, что устремляя h к нулю, невозможно получить численно сколь угодно точное значение производной. Чтобы получить оптимальное значение шага, при котором полная ошибка будет минимальна, нужно решать оптимизационную задачу.

Задача 1 (оптимальное значение шага). *Найти оптимальный шаг h при вычислении производной по формуле* (1) *на компьютере с машинным эпсилон ϵ_0, если функция f принадлежит классу функций с ограниченной второй производной $|f''(x)| \le M_2$, а модуль функции в окрестности x_0, на которой выбирается x_1, не превосходит M.*

Решение: полная ошибка ϵ в задаче будет состоять из ошибки метода ϵ_{method} и ошибки округления ϵ_{trunc}. Оптимальным шагом будет тот, на котором достигается минимум оценки полной ошибки. Оценка ошибки округления получается по формуле (3) с учетом ограниченности функции f в окрестности x_0: $\epsilon_{trunc} \le \frac{2M\epsilon_0}{h}$.

Оценим ошибку метода. Для этого в формуле (1) разложим $f(x_1)$ по формуле Тейлора относительно точки x_0 и используем форму Лагранжа для остаточного члена:

$$f'(x_0) \approx \frac{f_1 - f_0}{h} = f'(x_0) + \frac{h}{2}f''(x_0) + O(h^2) = f'(x_0) + \frac{h}{2}f''(\xi).$$

Отсюда видно, что старший член ошибки метода $\frac{h}{2}f''(x_0)$ и оценка ошибки метода: $\epsilon_{method} \le \frac{h}{2}|f''(\xi)| \le \frac{hM_2}{2}$. Здесь учли, что функция f принадлежит классу функций с ограниченной второй производной. Тогда полная ошибка $\epsilon \le \frac{hM_2}{2} + \frac{2M\epsilon_0}{h}$. Минимизируя полную ошибку по h (приравняем производную по h к нулю), получим оптимальное значение шага $h_{opt} = 2\sqrt{\frac{M\epsilon_0}{M_2}}$. ■

2.5. Абсолютная и относительная ошибки

Пусть при вычислении функции было получено значение $\tilde{z}$, а ее точное значение z. Тогда абсолютная ошибка вычисления функции будет равна $|z - \tilde{z}|$, а относительная ошибка – $\left|\frac{z - \tilde{z}}{\tilde{z}}\right|$.

Задача 2 (абсолютная и относительная ошибка).
Величина x задана с ошибкой δx. Оцените относительную и абсолютную ошибку при вычислении функции $f = f(x)$.

Решение: Абсолютная ошибка получается из оценки $\delta f \le$ $\le |f'(x)|\,\delta x$. Тогда относительная ошибка $\frac{\delta f}{|f|} \le |f'(x)|\,\frac{\delta x}{|f|}$. ■

3. Численное дифференцирование

3.1. Метод неопределенных коэффициентов

Пусть в одномерной области $[x_{\min}, x_{\max}]$ задана равномерная сетка из $N+1 = m+l+1$ узлов (Равномерная сетка – сетка, расстояние между двумя любыми соседними узлами которой равно h, где h – сеточный шаг). На этой области определена бесконечно непрерывно дифференцируемая функция f. Известны значения этой функции во всех узлах рассматриваемой сетки $\{f_i\}_{i=-l}^{m}$. В этом случае говорят, что определена *сеточная функция* – проекция функции на сетку. Пусть необходимо вычислить значение производной в некотором узле j, слева от которого l узлов, справа m. Построим метод максимального порядка точности по значениям функции в сеточных узлах. Под порядком точности метода понимаем степень при h старшего члена ошибки.

Для этого представим производную в узле j как сумму значений функции во всех узлах, взятых с некоторыми весами:

$$f'(x_j) \approx \frac{1}{h} \sum_{k=-l}^{m} \alpha_k f(x_j + kh). \tag{4}$$

Подберем веса так, чтобы по этим значениям функции порядок точности метода был максимальным. Оказывается, что по $N+1$ точке можно построить метод N-го порядка точности. Для этого разложим в ряд Тейлора все члены, входящие в суммирование в (4), относительно точки x_j, сгруппируем члены при одинаковых степенях и приравняем к нулю коэффициенты при степенях ниже $N+1$ (кроме первой, для нее приравняем к 1). В итоге получим $N+1$ уравнение относительно $N+1$ неизвестной. В матричном виде получившуюся систему можно представить как $A\alpha = b$, где $b^T = (0, 1, 0, ..., 0)^T$, а матрица A:

$$A = \begin{pmatrix} 1 & 1 & ... & 1 \\ -l & -l+1 & ... & m \\ (-l)^2 & (-l+1)^2 & ... & m^2 \\ (-l)^3 & (-l+1)^3 & ... & m^3 \\ ... & ... & ... & ... \end{pmatrix} \tag{5}$$

является матрицей Вандермонда, поэтому система всегда имеет единственное решение.

Задача 3 (МНК для численного дифференцирования). *Задана табличная функция*

x	-1	1	2
$f(x)$	5	2	1

Функция $f(x)$ во всех узлах задана с абсолютной погрешностью 10^{-1}. Пусть функция $f(x)$ принадлежит классу функций: $\max\left|f^{(3)}(x)\right| \le M_3 = 0.3$. Найти формулу вычисления производной в точке $x = -1$ со вторым порядком аппроксимации, вычислить производную и оценить точность вычисленного значения производной.

Решение: аппроксимационная формула для вычисления первой производной с помощью метода неопределенных коэффициентов будет иметь вид

$$f'(x) \approx \frac{\alpha_0 f(x) + \alpha_1 f(x+2h) + \alpha_2 f(x+3h)}{h}. \qquad (6)$$

Здесь $h = 1$. Подставим в эту формулу разложение в ряд Тейлора функций $f(x+2h)$ и $f(x+3h)$ до членов третьего порядка по h (сколько узлов, до такого порядка и надо разложить). Сгруппировав члены по степеням h, получим систему уравнений на неопределенные коэффициенты. Решая систему, получим

$$\alpha_0 = -\frac{5}{6}, \alpha_1 = \frac{3}{2}, \alpha_2 = -\frac{2}{3}.$$

Ошибку входных данных оценим по аналогии с тем, как это делалось в формуле (2):

$$\Delta_{in} = \frac{|\alpha_0\delta_0| + |\alpha_1\delta_1| + |\alpha_2\delta_2|}{h} = 0.3.$$

Ошибка метода – те члены, которые не обнулились после подстановки разложений в ряд Тейлора $f(x+2h)$ и $f(x+3h)$ в формуле (6) за счет выбора неопределенных коэффициентов:

$$\Delta_{method} = \frac{|\alpha_1 f'''(\xi_1)|\,\frac{8h^3}{6} + |\alpha_2 f'''(\xi_2)|\,\frac{27h^3}{6}}{h} \le M_3 5h^2 = 1.5.$$

Полная ошибка – сумма ошибки метода и входных данных (≤ 1.8). ■

3.2. Оценка порядка точности метода

Рассмотрим метод с порядком точности p. Тогда ошибка метода $\epsilon_h = Ch^p$, где h – сеточный шаг. На сетке с вдвое меньшим шагом ошибка метода будет $\epsilon_{h/2} = C_1 \left(\frac{h}{2}\right)^p$. Если шаг h достаточно мелкий (функция меняется не очень сильно), то можно считать, что $C \approx C_1$. Тогда, исключив C из первого равенства за счет второго, можно получить, что

$$p = \log_2 \frac{\epsilon_h}{\epsilon_{h/2}}. \tag{7}$$

4. Численные методы решения СЛАУ

Численные методы решения систем линейных алгебраических уравнений (СЛАУ) условно можно поделить на два класса: *прямые методы* и *итерационные методы*. Если вычисления не ограничены машинной точностью, то прямые методы позволяют находить решение точно после выполнения заданного количества элементарных операций, определяемых методом. Итерационные методы дают приближенное решение с любой заданной точностью после некоторого количества итераций, определяемого желаемой точностью. Классические примеры прямых методов: метод Гаусса, LU-разложение, разложение Холецкого (метод квадратного корня), метод трехдиагональной прогонки. Выбор метода зависит от постановки задачи. Какие-то методы применяются к системам с матрицами общего вида, другие – специального. Классические примеры итерационных методов: метод простой итерации, метод Якоби, метод Зейделя.

4.1. Нормы и обусловленность

Векторные нормы:

$$\|u\|_\infty = \max_i |u_i|,$$
$$\|u\|_1 = \sum_i |u_i|,$$
$$\|u\|_2 = \left(\sum_i |u_i|^2 \right)^{\frac{1}{2}}.$$

Матричные нормы:

$$\|A\|_\infty = \max_i \sum_j |a_{ij}|,$$
$$\|A\|_1 = \max_j \sum_i |a_{ij}|,$$
$$\|A\|_2 = \left(\max_i \lambda_i(A^* A) \right)^{\frac{1}{2}} = \max_i \sigma_i(A).$$

Здесь σ_i – сингулярное число матрицы.

В общем случае векторная и матричная норма связаны следующим соотношением (определение подчиненной нормы):

$$\|A\| = \sup_u \frac{\|Au\|}{\|u\|}.$$

Важно помнить, что матричные нормы удовлетворяют свойству:

$$\|AB\| \leq \|A\|\|B\|$$

для всех матриц A и B допускающих умножение, в том числе и когда B – это вектор-столбец.

Задача 4 (проверка на выполнение свойств нормы). *Показать, что для вектора* $\mathbf{x} = (x_1, x_2)^T$ *выражение* $\|x\|_* = \max(|x_1 - x_2|, |x_2|)$ *является нормой. Найти норму матрицы*

$$A = \begin{pmatrix} 2 & 0 \\ 1 & 9 \end{pmatrix},$$

подчиненную этой векторной норме.

Решение: заметим, что $\|x\|_* = \|Sx\|_\infty$, где

$$S = \begin{pmatrix} 1 & -1 \\ 0 & 1 \end{pmatrix}.$$

Используя это свойство, несложно убедиться, что все аксиомы нормы выполняются. Вычислим

$$\|A\|_* = \sup_x \frac{\|Ax\|_*}{\|x\|_*} = \sup_x \frac{\|SAx\|_\infty}{\|Sx\|_\infty}.$$

Далее используем замену $Sx = y$, тогда

$$\|A\|_* = \sup_y \frac{\|SAS^{-1}y\|_\infty}{\|y\|_\infty} = \|SAS^{-1}\|_\infty = 11. \blacksquare$$

Обусловленность СЛАУ – то, как будет меняться решение системы при возмущении матрицы системы и ее правой части. Иными словами – чувствительность к ошибкам на входе (в частности, к ошибкам округления):

$$(A+\delta A)(u+\delta u) = f+\delta f,$$
$$\frac{||\delta u||}{||u||} \le \frac{\mu}{1-\mu\frac{||\delta A||}{||A||}}\left(\frac{||\delta f||}{||f||}+\frac{||\delta A||}{||A||}\right).$$

Здесь δA – возмущение матрицы системы, δf – возмущение правой части, μ – *число обусловленности матрицы* A,

$$\mu(A) = ||A^{-1}|| \cdot ||A||, \mu \ge 1.$$

Число обусловленности можно считать мерой обусловленности системы.

Задача 5 (оценка относительной ошибки решения системы). *Для СЛАУ* $Ax = f$, *где* $f = (1, f_2)^T$ *и*

$$A = \begin{pmatrix} 2 & 1 \\ 1 & 2 \end{pmatrix}$$

матрица A *задана точно, а правая часть* f *может иметь возмущение* δf. *Найти такое* f_2, *при котором выполняется неравенство* $\frac{\|\delta x\|_2}{\|x\|_2} \le \frac{\|\delta f\|_2}{\|f\|_2}$.

Решение: рассмотрим соотношение $\frac{\|\delta x\|_2}{\|x\|_2} \le \nu(f)\frac{\|\delta f\|_2}{\|f\|_2}$. Чтобы удовлетворить условию задачи, надо найти такую f, чтобы $\nu(f) = 1$. Найдем верхнюю грань $\nu(f)$ по всем возмущениям правой части δf:

$$\nu(f) = \sup_{\delta f}\frac{\|\delta x\|_2}{\|x\|_2}\frac{\|f\|_2}{\|\delta f\|_2} = \sup_{\delta f}\frac{\|f\|_2}{\|x\|_2}\frac{\|A^{-1}\delta f\|_2}{\|\delta f\|_2} = \frac{\|f\|_2}{\|x\|_2}\|A^{-1}\|_2.$$

Чтобы получить последнее равенство, было использовано определение нормы матрицы $\|A^{-1}\|_2 = \sup_{\delta f}\frac{\|A^{-1}\delta f\|_2}{\|\delta f\|_2}$. Несложно убедиться, что в этой задаче $\|A^{-1}\|_2 = 1$. Решая исходную СЛАУ, получим решение x относительно f_2, $x = \frac{1}{3}(2-f_2, 2f_2-1)^T$. Приравнивая $\nu(f)$ к единице, из последнего равенства получаем соотношение на f_2 для выполнения требуемой оценки. А именно: $\frac{\sqrt{1+f_2^2}}{\sqrt{(2-f_2)^2/9+(2f_2-1)^2/9}}\cdot 1 = 1$. Решение этого уравнения дает искомое значения $f_2 = -1$. ■

Замечание: можно показать, что верхняя грань $\nu(f)$ по f как раз равняется μ, а нижняя – одному. То есть провести более строгую оценку в задаче не выйдет.

4.2. Прямые методы

При решении системы вида $Ax = f$ после применения метода Гаусса конечный вид системы после преобразований будет $Ux = c$, после применения LU-разложения – $LUx = f$. Здесь U – верхнетреугольная матрица, L – нижнетреугольная матрица с единицами на диагонали. Метод Гаусса и метод, основанный на LU-разложении, могут быть применены к системам с матрицами общего вида. Элементарные операции, необходимые для представления матрицы A в виде $A = LU$, аналогичны элементарным операциям, используемым в методе Гаусса. В случае, когда матрица A симметричная и положительно определенная, может быть использован метод квадратного корня (разложение Холецкого) $LL^Tx = f$. Здесь элементы матрицы L вычисляются по формулам. Симметричные и положительно определенные матрицы часто возникают, например, при использовании метода наименьших квадратов и численном решении дифференциальных уравнений. По сравнению с методом Гаусса или LU-разложением, он устойчивее численно и требует примерно вдвое меньше арифметических операций.

Решение методом Гаусса получается применением обратного хода по конечному виду системы $Ux = c$. Начиная с последней строки и двигаясь последовательно к первой находятся все неизвестные системы. Схема решения задачи с помощью LU-разложения сводится к последовательному решению двух систем $Ly = f$ (прямой ход) и $Ux = y$ (обратный ход). Аналогично и для метода квадратного корня. Стоит обратить внимание, что методы, основанные на разложении матрицы системы, в отличии от метода Гаусса, не меняют правую часть системы. Это означает, что они могут быть применены многократно для систем с одной и той же матрицей и разными правыми частями. Такая задача возникает, например, при численном обращении матрицы.

При применении метода Гаусса стоит помнить о выделении главного члена. Выделение главного члена позволяет избежать проблем, связанных с округлением при численном решении зада-

чи. Хотя выделение главного члена может быть не всегда оправдано, так как может нарушать структуру матрицы системы.

В случае, когда для всех строк матрицы системы выполняется *условие диагонального преобладания*

$$|a_{ii}| \geq \sum_{i \neq j} |a_{ij}|,$$

выделять главный член не нужно.

Задача 6 (разложение Холецкого). *Представить матрицу*

$$A = \begin{pmatrix} 4 & -2 & 2 \\ -2 & 2 & -4 \\ 2 & -4 & 11 \end{pmatrix}$$

в виде $A = LL^T$, *используя разложение Холецкого.*

Решение: матрица L имеет общий вид

$$L = \begin{pmatrix} L_{11} & 0 & 0 \\ L_{21} & L_{22} & 0 \\ L_{31} & L_{32} & L_{33} \end{pmatrix}.$$

Тогда справедливо

$$A = \begin{pmatrix} L_{11}^2 & L_{11}L_{21} & L_{11}L_{31} \\ L_{11}L_{21} & L_{21}^2 + L_{22}^2 & L_{21}L_{31} + L_{22}L_{32} \\ L_{11}L_{31} & L_{21}L_{31} + L_{22}L_{32} & L_{31}^2 + L_{32}^2 + L_{33}^2 \end{pmatrix}.$$

Используя такое представление матрицы A будем последовательно вычислять элементы матрицы L, двигаясь по матрице A от первой строки к последней, начиная каждый раз с диагонального элемента вправо: $L_{11} = \sqrt{4} = 2$, $L_{21} = -2/L_{11} = -1$, $L_{31} = 2/L_{11} = 1$, $L_{22} = \sqrt{2 - L_{21}^2} = 1$, $L_{32} = \dfrac{-4 - L_{21}L_{31}}{L_{22}} = -3$, $L_{33} = \sqrt{11 - L_{31}^2 - L_{32}^2} = 1$.

4.3. Метод простой итерации (метод Ричардсона)

Для решения СЛАУ вида $Ax = b$ метод простой итерации будет иметь вид

$$x_{k+1} = (E - \tau A)x_k + \tau b, \tag{8}$$

где k – итерационный индекс, τ – скалярный итерационный параметр. Стоит указать, что метод применим для знакоопределенных самосопряженных матриц.

Задача 7 (метод простой итерации для решения СЛАУ). *Для СЛАУ* $Ax = b$, *где* $b = (6, -6, 1)^T$ *и*

$$A = \begin{pmatrix} 6 & 0 & 7 \\ 0 & 6 & 6 \\ 7 & 6 & 18 \end{pmatrix}$$

1. *вычислить число обусловленности в трех нормах;*

2. *для заданной относительной погрешности правой части* $\frac{\|\delta b\|_2}{\|b\|_2} = 0.01$ *найти границы для относительной погрешности* $\frac{\|\delta x\|_2}{\|x\|_2}$ *решения системы;*

3. *исследовать на сходимость и оценить скорость сходимости МПИ* (8) *при* $\tau = 0.01$*;*

4. *найти оптимальный параметр МПИ* τ *и дать оценку сходимости при этом* τ.

Замечание: заметим, что в данном случае матрица A *самосопряженная, поэтому удобно считать вторую норму матрицы. В противном случае можно было бы домножить систему слева на* A^T.

Решение:

1. Вычисляется по формулам выше. Во второй норме число обусловленности $\mu_2 = \dfrac{\lambda_{\max}(A)}{\lambda_{\min}(A)} = 23$, так как $\lambda_{\max}(A^{-1}) = \dfrac{1}{\lambda_{\min}(A)}$. $\lambda_{\max}$ и $\lambda_{\min}$ – максимум и минимум модуля собственного числа матрицы соответственно.

2. По формуле, связывающей относительную ошибку, ошибку входных данных и число обусловленности можно получить

$$\frac{||\delta x||_2}{||x||_2} \leq \mu_2 \frac{||\delta b||_2}{||b||_2} = 23 \cdot 0.01 = 0.23.$$

3. Для симметричной матрицы скорость сходимости можно определить как $q = ||B||_2 = ||E - \tau A||_2 = \max_i |1 - \tau\lambda_i|$. Условие сходимости $q < 1$ сводится к решению системы:

$$\begin{cases} |1 - \tau\lambda_{\max}| < 1, \\ |1 - \tau\lambda_{\min}| < 1, \end{cases}$$

решение которой дает условие сходимости $0 < \tau < \dfrac{2}{\lambda_{\max}}$. Скорость сходимости при $\tau = 0.01$:

$$q = \max\left(|1 - 0.01 \cdot 1|, |1 - 0.01 \cdot 6|, |1 - 0.01 \cdot 23|\right) = 0.99.$$

4. Для симметричной положительно определенной матрицы оптимальный итерационный параметр

$$\tau_{opt} = \frac{2}{\lambda_{\max} + \lambda_{\min}} = \frac{1}{12},$$

скорость сходимости

$$q = \frac{\lambda_{\max} - \lambda_{\min}}{\lambda_{\max} + \lambda_{\min}} = \frac{11}{12}.$$

Замечание: оценку числа итераций k, необходимых для сходимости итерационного процесса, можно получить по формуле

$$k \geq \frac{\ln \epsilon/\epsilon_0}{\ln q},$$

где ϵ – желаемая точность, ϵ_0 – норма разности между точным решением и начальным приближением (ошибка начального приближения), q – скорость сходимости. В учебных задачах точное решение как правило известно и ϵ_0 можно вычислить явно. На практике точное решение не известно и ϵ_0 оценивается исходя из конкретной информации о рассматриваемой задаче. ■

Замечание: оптимальный параметр находится из условия $\min_\tau \max_i |1 - \tau\lambda_i|$ и не учитывает правую часть решаемой системы. Поэтому скорость сходимости при выборе оптимального параметра будет достаточно хорошей, но не наилучшей.

Замечание: в методе Гаусса без выбора главного члена при больших размерах матрицы n требуемое число операций для решения системы $\approx \frac{2}{3}n^3$, в МПИ $\approx 2n^2 \cdot I$, где I – число итераций. То есть при $I < \frac{n}{3}$ МПИ лучше. Обычно $I \ll n$.

Замечание: часто рассмотренный метод называют именно метод Ричардсона, потому что МПИ – это любой метод вида $x^{k+1} = Sx^k + b$.

Замечание: погрешность, вносимая в решение из-за конечной разрядности мантиссы не зависит от количества итераций.

Задача 8 (сходимость метода простой итерации). *При каких положительных α и β сходится МПИ $x_{k+1} = Bx_k + f$, где*

$$B = \begin{pmatrix} \alpha & \beta & 0 \\ \beta & \alpha & \beta \\ 0 & \beta & \alpha \end{pmatrix}$$

и $\alpha > \sqrt{2}\beta$.

Решение: критерий сходимости МПИ – все собственные числа матрицы B по модулю меньше единицы. Найдем собственные числа матрицы: $\lambda_1 = \alpha, \lambda_{2,3} = \alpha \pm \sqrt{2}\beta$. Так как α и β положительные и $\alpha > \sqrt{2}\beta$, то максимальное по модулю собственное число $\lambda = \alpha + \sqrt{2}\beta$. Тогда все подходящие α и β находятся из условия $|\alpha + \sqrt{2}\beta| < 1$. ■

Замечание: достаточное условие сходимости метода простой итерации: $\|B\| \leq q < 1$. Использование достаточного условия в предыдущей задаче вместо критерия может дать не все возможные значения α и β, при которых МПИ будет сходиться.

4.4. Методы Якоби и Зейделя

Решается система $A\mathbf{u} = f$, $A = L + D + U$; L, U, D – нижнетреугольная, верхнетреугольная и диагональная матрицы соответственно. Покомпонентно расчетные формулы метода Якоби будут иметь вид

$$\begin{aligned} u_1^{k+1} &= -(a_{12}u_2^k + a_{13}u_3^k + \ldots + a_{1n}u_n^k - f_1)/a_{11}, \\ u_2^{k+1} &= -(a_{21}u_1^k + a_{23}u_3^k + \ldots + a_{1n}u_n^k - f_2)/a_{22}, \\ &\ldots \\ u_n^{k+1} &= -(a_{n1}u_1^k + a_{n2}u_2^k + \ldots + a_{n,n-1}u_{n-1}^k - f_n)/a_{nn}. \end{aligned}$$

В матричном виде метод Якоби:

$$\mathbf{u^{k+1}} = -D^{-1}(A - D)\mathbf{u^k} + D^{-1}f.$$

Для метода Зейделя:

$$\begin{aligned}
u_1^{k+1} &= -(a_{12}u_2^k + a_{13}u_3^k + \ldots + a_{1n}u_n^k - f_1)/a_{11},\\
u_2^{k+1} &= -(a_{21}u_1^{k+1} + a_{23}u_3^k + \ldots + a_{1n}u_n^k - f_2)/a_{22},\\
&\ldots\\
u_n^{k+1} &= -(a_{n1}u_1^{k+1} + a_{n2}u_2^{k+1} + \ldots + a_{n,n-1}u_{n-1}^{k+1} - f_n)/a_{nn}.
\end{aligned}$$

В матричном виде метод Зейделя:

$$\mathbf{u^{k+1}} = -(L + D)^{-1}U\mathbf{u^k} + (L + D)^{-1}f.$$

Задача 9 (сходимость метода Якоби). *Найти условия сходимости метода Якоби для численного решения СЛАУ вида* $Ax = d$*, где*

$$A = \begin{pmatrix} \alpha & 0 & \beta \\ 0 & \alpha & 0 \\ \beta & 0 & \alpha \end{pmatrix}.$$

Используйте критерий сходимости.

Решение: проверим критерий сходимости метода Якоби. Для этого найдем λ из условия

$$\det \begin{pmatrix} \lambda\alpha & 0 & \beta \\ 0 & \lambda\alpha & 0 \\ \beta & 0 & \lambda\alpha \end{pmatrix} = 0$$

и проверим условие $|\lambda| < 1$. Вычисляя детерминант, получим $\lambda_1 = 0, \lambda_{2,3} = \pm\frac{\beta}{\alpha}$. Отсюда условие сходимости $|\beta| < |\alpha|$. ■

Замечание: если в критерии для метода Якоби на λ домножалась только диагональ матрицы A, то для метода Зейделя домножать нужно всю ее нижнетреугольную часть. В остальном условие то же.

Задача 10 (оценка числа итераций для метода Якоби). *Оцените количество итераций для достижения точности*

$\epsilon = 10^{-6}$ в норме $||\cdot||_\infty$ при решении итерационным методом Якоби линейной системы размера n с трехдиагональной действительной матрицей A вида

$$A = \begin{bmatrix} 1 & a_1 & 0 & \dots \\ b_2 & 2 & a_2 & \dots \\ & \ddots & \ddots & \ddots \\ & & b_n & n \end{bmatrix},$$

если известно, что $|a_i| \le i/5$, $|b_i| \le i/10$, а $\|x\|_\infty = 15$ (x – точное решение).

Решение: можно оценить норму итерационной матрицы, $||D^{-1}(A-D)||_\infty$.

В этой матрице на диагонали нули, а внедиагональные элементы в каждой строке поделены на соответствующий диагональный элемент (так работает умножение на диагональную матрицу слева). $||\cdot||_\infty$ – это максимальная сумма модулей по строке, поэтому

$$||D^{-1}(A-D)||_\infty \le \max_i \frac{|a_i| + |b_i|}{i} \le \frac{3}{10}.$$

Возьмем нулевой вектор в качестве начального приближения, тогда

$$\|e_0\|_\infty = \|x - x_0\|_\infty = 15.$$

Оценку числа итераций можно найти по общей формуле

$$||e^k|| \le q^k ||e_0|| \le \epsilon \Rightarrow k \ge \frac{\ln(\epsilon/||e_0||)}{\ln(q)} = \frac{\ln(10^{-6}/15)}{\ln(\frac{3}{10})} \approx 13.7. \blacksquare$$

4.5. Численные методы решения переопределенных СЛАУ

Предположим, что требуется представить функцию $f(x)$ как линейную комбинацию некоторых известных (базисных) функций с постоянными коэффициентами:

$$f(x) = a_1 f_1(x) + a_2 f_2(x) + ... + a_m f_m(x), \tag{9}$$

при этом известны значения функции $f(x_i) = y_i$ для n узлов. Также пусть $n > m$. В этом случае система на коэффициенты

a_i, $F\mathbf{a} = y$, будет переопределенной. Несложно убедиться, что i-я строка будет состоять из значений базисных функций, взятых в узле x_i. Сформулированная задача – *задача линейной регрессии.* Для ее решения можно использовать метод, называемый *Методом наименьших квадратов (МНК).* Идея метода – подобрать коэффициенты так, чтобы решение было наилучшим в смысле минимизации среднеквадратичного отклонения по всем узлам:

$$S = \sum_{i=1}^{n} \left(y_i - \sum_{j=1}^{m} a_j f_j(x_i) \right)^2 = \|\mathbf{y} - F\mathbf{a}\|_2^2 . \tag{10}$$

Можно показать, что минимизация S эквивалентна решению системы вида $F^T F a = F^T y$, решение которой и даст значения коэффициентов a_i.

Замечание: в машинном обучении при решении задачи линейной регрессии (9) неизвестные коэффициенты a_i будут называться весами, базисные функции $f_i(x)$ – признаками, S – функцией потерь. Формулировка задачи при этом остается прежней.

Задача 11 (решение переопределенной СЛАУ). *Решите переопределенную СЛАУ, используя теорему о методе наименьших квадратов:*

$$\begin{cases} x + y = 1, \\ 2x + y = 2, \\ x + 3y = 5. \end{cases}$$

Решение: составим и решим систему $F^T F\mathbf{u} = F^T b$, где $\mathbf{u} = (x, y)^T$. Несложно проверить, что $F^T F = \begin{pmatrix} 6 & 6 \\ 6 & 11 \end{pmatrix}$, $F^T b = (10, 18)^T$. Решая полученную систему, например, методом Гаусса, получим решение в смысле метода наименьших квадратов: $(x, y)^T = \frac{1}{30}(2, 48)^T$. ■

Если базисные функции линейно зависимы, то матрица $F^T F$ будет вырождена (в машинном обучении в этом случае говорят, что признаки скоррелированы). Это будет означать, что задача минимизации (10) будет иметь неединственное решение. Чтобы избежать этой проблемы, прибегают к регуляризации (10). До-

бавим к (10) функцию штрафа $\alpha\|\mathbf{a}\|_2^2$ и будем теперь минимизировать функцию

$$S = \|\mathbf{y} - F\mathbf{a}\|_2^2 + \alpha\|\mathbf{a}\|_2^2 . \tag{11}$$

По сути это означает, что среди всех возможных весов будет выбран один, который минимизирует функцию штрафа. Можно показать, что решение задачи (11) эквивалентно решению задачи $(F^TF + \alpha I)\mathbf{a} = F^T\mathbf{y}$, где матрица $F^TF + \alpha I$ уже будет невырожденной, I – единичная матрица. Регуляризация вида (11), где функция штрафа берется во второй норме, называется *Тихоновская регуляризация* (ridge regression). Регуляризация может быть использована не только для разрешения случаев с линейно зависимыми базисными функциями, но и тогда, когда данные сильно зашумлены или в данных присутствуют нефизичные выбросы. В качестве примера можно также привести регуляризацию lasso, где функция штрафа берется в первой норме и Elastic Net – комбинация lasso и ridge regression.

5. Интерполяция

Интерполяция – это задача восстановления непрерывной функции по дискретному набору ее значений. Этот набор значений можно назвать *сеточная функция*. Сеточная функция – это проекция функции на сетку. Задача интерполяции неоднозначна. Восстановленную функцию называют *интерполянтом*. Главное требование к интерполянту – он должен совпадать в сеточных узлах с сеточной функцией. В качестве примеров способов интерполяции можно привести полиномиальную интерполяцию, интерполяцию сплайнами, кусочно-линейную интерполяция (которая строго говоря тоже является интерполяцией сплайнами). Самый простой способ из представленных – кусочно-линейная интерполяция. В случае кусочно-линейной интерполяции функция восстанавливается посредством соединения соседних точек линейным образом. Часто в практических задач кусочно-линейной интерполяции оказывается достаточно. Например, в случае, когда интерполяция производится на очень подробной сетке. Далее рассмотрим два других примера – полиномиальную интерполяцию и интерполяцию сплайнами.

5.1. Полиномиальная интерполяция

Пусть в одномерной области $[x_{\min}, x_{\max}]$ задана равномерная сетка из $N+1$ узла. На этой области определена бесконечно непрерывно дифференцируемая функция f. Известны значения этой функции во всех узлах рассматриваемой сетки $\{f_i\}_{i=0}^N$. Будем искать интерполянт в следующем виде:

$$F(x) = \sum_{i=0}^{N} u_i \phi_i(x). \tag{12}$$

$\phi_i(x)$ – базисные функции, u_i – некоторые постоянные коэффициенты. Если базис состоит из полиномов, например, $\{\phi_i(x)\}_{i=0}^N = \{x^i\}_{i=0}^N$, то тогда говорят, что определена задача полиномиальной интерполяции. Чтобы ее решить, необходимо найти коэффициенты $\{u_i\}_{i=0}^N$. Находятся они из условия, что интерполянт должен совпадать в узлах с сеточной функцией. Тогда получим

следующую систему на неизвестные коэффициенты:

$$\begin{cases} u_0 + u_1 x_0 + u_2 x_0^2 + \ldots + u_N x_0^N = f_0, \\ u_0 + u_1 x_1 + u_2 x_1^2 + \ldots + u_N x_1^N = f_1, \\ \ldots \\ u_0 + u_1 x_N + u_2 x_N^2 + \ldots + u_N x_N^N = f_N. \end{cases} \tag{13}$$

Решая данную СЛАУ, находим коэффициенты интерполяционного полинома. В матричном виде получившуюся систему можно представить как $Au = b$, где

$$b = (f_0, f_1, \ldots, f_N)^T, \tag{14}$$

$$A = \begin{pmatrix} 1 & x_0 & \ldots & x_0^N \\ 1 & x_1 & \ldots & x_1^N \\ 1 & x_2 & \ldots & x_2^N \\ \ldots & \ldots & \ldots & \ldots \\ 1 & x_N & \ldots & x_N^N \end{pmatrix}. \tag{15}$$

A является матрицей Вандермонда, поэтому система всегда имеет единственное решение. Однако, матрица Вандермонда плохо обусловлена, для любой сетки число обусловленности растет примерно как $2^N/\sqrt{N}$. Поэтому бессмысленно строить интерполяционный полином таким способом по большому количеству узлов.

В конечном счёте нам нужны не коэффициенты интерполяционного полинома, а какой-то способ вычисления его значения в точке. В следующих пунктах описаны две формы полинома, которые позволяют это сделать.

5.2. Построение интерполяционного полинома. Форма Лагранжа и Ньютона

При построении интерполяционного полинома, удобно его представлять в одной из следующих форм:

Форма Лагранжа

$$L(x) = \sum_{i=0}^{N} f_i \psi_i(x),$$

где $f_i = f(x_i)$, а $\psi_i(x) = \prod\limits_{j=0, i\neq j}^{N} \dfrac{x - x_j}{x_i - x_j}$.

$\psi_i(x)$ – это базовые полиномы, которые устроены очень просто: в i-м узле они равны 1, а во всех остальных узлах – 0. Если запомнить это свойство, то можно легко вывести формулу для них.

Представление полинома в форме Лагранжа позволяет избежать непосредственного решения СЛАУ (13).

Форма Ньютона. Представление интерполяционного полинома в форме Ньютона удобно при решении учебных задач и также позволяет избежать решения СЛАУ (13). Она имеет следующий вид:

$$N(x) = f(x_0) + f(x_0, x_1)(x - x_0) + \ldots + f(x_0, \ldots, x_N)(x - x_0)\ldots(x - x_{N-1}), \tag{16}$$

где $f(x_i, \ldots, x_{i+j})$ – разделенные разности. Значения сеточной функции в узлах сетки $f(x_i)$ называют разделенными разностями нулевого порядка, $f(x_i, x_{i+1}) = \dfrac{f(x_{i+1}) - f(x_i)}{x_{i+1} - x_i}$ – разделенные разности первого порядка, $f(x_i, x_{i+1}, x_{i+2}) = \dfrac{f(x_{i+1}, x_{i+2}) - f(x_i, x_{i+1})}{x_{i+2} - x_i}$ – разделенные разности второго порядка и т.д. Разделенные разности порядка k считаются по разделенным разностям порядка $k - 1$.

Задача 12 (интерполяция квадратичной функцией).
Задана табличная функция

x	0	1	2
$f(x)$	0	1	4

По табличной функции построить интерполяционный полином, используя форму Лагранжа и форму Ньютона.

Решение: построим сначала в форме Лагранжа:

$$L(x) = 0 \cdot \frac{(x-1)(x-2)}{(0-1)(0-2)} + 1 \cdot \frac{(x-0)(x-2)}{(1-0)(1-2)} + 4 \cdot \frac{(x-0)(x-1)}{(2-0)(2-1)} = x^2.$$

Теперь построим в форме Ньютона:

$$N(x) = f(x_0) + f(x_0, x_1)(x - x_0) + f(x_0, x_1, x_2)(x - x_0)(x - x_1).$$

Чтобы найти разделенные разности $f(x_0, x_1)$ и $f(x_0, x_1, x_2)$, составим таблицу разделенных разностей. В первом столбце сеточные узлы, далее – разделенные разности нулевого, первого и второго порядков:

x_i	$f(x_i)$	$f(x_i, x_{i+1})$	$f(x_i, x_{i+1}, x_{i+2})$
0	0	1	1
1	1	3	
2	4		

Необходимые разделенные разности представлены в первой строчке. Тогда $N(x) = 0 + 1(x - 0) + 1(x - 0)(x - 1) = x^2$. ■

Замечание: используя разные формы, получается один и тот же полином. Этого и следовало ожидать, так как по построению интерполяционный полином единственен. Форма Лагранжа и форма Ньютона – две разные формы представления одного и того же полинома.

Задача 13 (обратная интерполяция). *Задана табличная функция*

x	*1*	*2*	*3*	*4*	*5*
$f(x)$	*-9*	*-6*	*3*	*6*	*9*

С помощью полиномиальной интерполяции найти x, при котором $f(x) = 0$.

Решение: если подходить к задаче «в лоб», сразу строя интерполяционный полином, то в итоге для решения задачи придется найти корни уравнения четвертой степени, что является непростой задачей. При большем числе узлов задача может стать нерешаемой. Но если заметить, что с ростом x значения функции растут, то можно предположить, что данная сеточная функция соответствует некоторой монотонной функции. Вместе с предположением о ее непрерывной дифференцируемости это дает возможность предположить, что у нее существует обратная f^{-1}. Тогда можно воспользоваться обратной интерполяцией, считая, что $y = f(x)$ – независимая переменная, а сеточная функция, для которой строим интерполянт – $f^{-1}(y)$. В этом случае для решения задачи не надо решать уравнение четвертой степени, а достаточно подставить 0 в интерполяционный полином. То есть меняем

строки в таблице местами, строим интерполяционный полином для обратной функции (в любой из форм), подставляем туда 0 и получаем решение задачи. Здесь важно, что рассматриваемая функция оказалась взаимооднозначной. В противном случае так решить задачу не получится. ■

5.3. Учет производных при построении интерполяционного полинома. Форма Эрмита

Пусть вместе со значениями функции в узлах также известны некоторые значения ее производных. Потребуем, чтобы производные интерполяционного полинома в узлах совпадали с известными значениями производных сеточной функции. Для этого необходимо построить *интерполяционный полином в форме Эрмита*. Форма Эрмита в общем схожа с формой Ньютона (16). Отличие лишь в том, что при составлении таблицы разделенных разностей, каждая точка должна учитываться столько раз, сколько для нее известно производных плюс 1, а разделенная разность порядка k для одной и той же точки должна вычисляться по формуле

$$f(\underbrace{x_i, \ldots, x_i}_{\text{всего } k+1 \text{ раз}}) = \frac{f^{(k)}}{k!}. \tag{17}$$

Разберемся с этим на примере задачи.

Задача 14 (интерполяция с учетом производных в узлах). *Задана табличная функция*

x	*0*	*1*	*2*
$f(x)$	*0*	*1*	*4*
$f'(x)$	-	*2*	*4*
$f''(x)$	-	*2*	-

Для узла 0 *известно только значение функции, для узла* 1 *– значение функции, ее первой и второй производной, для узла* 2 *– значение функции и ее первой производной. По табличной функции построить интерполяционный полином, используя форму Эрмита.*

Решение: сначала составим таблицу разделенных разностей. Для этого точку 0 учтем в ней один раз, точку 1 – три раза, точку

2 – два раза. В таблице 0 pp – разделенная разность нулевого порядка и т.д.

x_i	0 pp	1 pp	2 pp	3 pp	4 pp	5 pp
0	0	1	1	0	0	0
1	1	**2**	**1**	0	0	
1	1	**2**	1	0		
1	1	3	1			
2	4	**4**				
2	4					

В таблице «**жирным**» выделены те разделенные разности, которые вычисляются по формуле (17). Первые такие разделенные разности совпадают со значением производной, а вторая разделенная разность $f(x_1, x_1, x_1) = \frac{2}{2!} = 1$. Отметим, что остальные значения разделенных разностей вычислялись по обычным формулам вычисления разделенных разностей (см. формулы под (16)). Интерполяционный полином будет иметь общий вид:

$N(x) = f(x_0) + f(x_0, x_1)(x - x_0) + f(x_0, x_1, x_1)(x - x_0)(x - x_1) + +f(x_0, x_1, x_1, x_1)(x - x_0)(x - x_1)^2 + f(x_0, x_1, x_1, x_1, x_2)(x - x_0) \times \times(x - x_1)^3 + f(x_0, x_1, x_1, x_1, x_2, x_2)(x - x_0)(x - x_1)^3(x - x_2)$.

Подставляя сюда вычисленные значения для разделенных разностей (значения из первой строки), получим: $N(x) = 0 + +1 \cdot (x - 0) + 1 \cdot (x - 0)(x - 1) = x^2$. ■

Замечание: из общего вида интерполяционного полинома видно, что максимальная степень полинома с учетом производных на 1 меньше, чем количество всех известных данных (значений функции и ее производных).

5.4. Остаточный член полиномиальной интерполяции

Функция

$$R(x) = f(x) - L(x)$$

называется *остаточным членом полиномиальной интерполяции.* Можно показать, что

$$R(x) = \frac{f^{(N+1)}(\xi)}{(N+1)!} \prod_{j=0}^{N} (x - x_j). \tag{18}$$

Здесь ξ – некоторая точка из области интерполяции. Из этой формулы понятно, почему в задаче об интерполяции квадратичной функции по трем узлам полином получается точно. Это следует из того, что в этой задаче $f^{(N+1)} = f^{(3)} = (x^2)''' \equiv 0$.

Замечание: важно понимать, что для интерполяционных полиномов в форме Лагранжа и в форме Ньютона для одной и той же самой задачи остаточный член будет одним и тем же.

Замечание: в случае полиномиальной интерполяции с учетом производных в узлах (форма Эрмита) остаточный член будет иметь вид

$$R(x) = \frac{f^{(M)}(\xi)}{M!} \prod_{j=0}^{N} (x - x_j)^{m_j},$$

где M – общее количество данных, m_j – число производных для данной точки плюс 1.

Задача 15 (ошибка полиномиальной интерполяции). *Оценить ошибку полиномиальной интерполяции функции $f(x) = \cos x$ на отрезке $[0, \pi/2]$ по трем равноотстоящим узлам.*

Решение: по условию задачи интерполяцию предлагается проводить по узлам $x_0 = 0$, $x_1 = \pi/4$ и $x_2 = \pi/2$. Чтобы оценить ошибку полиномиальной интерполяции ϵ на рассматриваемом отрезке, оценим остаточный член полиномиальной интерполяции на нем:

$$R(x) = \frac{f'''(\xi)}{3!}(x - x_0)(x - x_1)(x - x_2).$$

Тогда

$$\epsilon \leq \max_{[0,\pi/2]} |R(x)|.$$

То есть необходимо оценить третью производную функции и произведение $(x-x_0)(x-x_1)(x-x_2)$ по модулю. Производная по модулю на рассматриваемом отрезке ограничена, $|f'''(x)| \leq M_3 = 1$. Чтобы оценить произведение найдем его экстремумы. На концах отрезка произведение равно 0. Найдем локальные экстремумы. Для нахождения локальных экстремумов решим задачу $(x(x - \pi/4)(x - \pi/2))' = 0$. Отсюда можно получить, что произведение ≤ 0.186. Оценка производной и оценка произведения

после подстановки в оценку ошибки полиномиальной интерполяции дает ответ поставленной задачи $\epsilon \leq 0.032$. ■

5.5. Обусловленность задачи интерполяции

Значения в узлах интерполяции могут быть заданы с ошибкой. Важно знать, насколько построенный по этим данным интерполяционный полином отличается от полинома, построенного по правильным данным. Это можно легко оценить, используя форму Лагранжа.

Пусть f_i – это точные значения функции в узлах, а $\tilde{f}_i = f_i + \delta f_i$ – значения с погрешностью, по которым мы и строим интерполяционный многочлен. Тогда

$$L(x) = \sum_{i=0}^{N} f_i \psi_i(x); \tilde{L}(x) = \sum_{i=0}^{N} (f_i + \delta f_i)\psi_i(x) \Rightarrow$$

$$\Rightarrow \left|\tilde{L}(x) - L(x)\right| = \left|\sum_{i=0}^{N} \delta f_i \psi_i(x)\right| \leq \delta f \Lambda(x), \delta f = \max_i |\delta f|.$$

Функцию $\Lambda(x) = \sum_{i=0}^{N} |\psi_i(x)|$ называют *функцией Лебега*, а её максимум Λ на отрезке интерполяции – *константой Лебега*. Значение функции Лебега в точке, и, соответственно, константу Лебега можно рассчитать через базовые многочлены Лагранжа по узлам интерполяции.

Задача 16 (обусловленность задачи интерполяции).

x_i	0.	1.	2.
$f(x_i)$	1.	1.39	1.94

В таблице приведены значения некоторой функции f с погрешностями $\leq \delta = 5 \times 10^{-3}$. По этим значениям строится интерполяционный полином. Оценить погрешность в значении полинома в точке $x = 1.5$, если известно, что $\max_{x \in [0,2]} |f^{(k)}(x)| \leq$ $\leq 2/3^k$.

Решение: Обозначим опять $\tilde{L}(x)$ – значение интерполяционного многочлена, построенного по неточным значениям. Тогда

$$|f(x) - \tilde{L}(x)| = |f(x) - L(x) + L(x) - \tilde{L}(x)| \leq$$
$$\leq |f(x) - L(x)| + |L(x) - \tilde{L}(x)| = E_1(x) + E_2(x).$$

Первое слагаемое в ошибке оценим через остаточный член интерполяции:

$$E_1(x = 1.5) = \left| \frac{f^{(3)}(\xi)}{3!}(x - x_0)(x - x_1)(x - x_2) \right| \leq$$
$$\leq \left| \frac{2}{3^3\, 3!}(1.5) \cdot (0.5) \cdot (-0.5) \right| \approx 4.6 \cdot 10^{-3}.$$

Вторая часть ошибки E_2 связана с обусловленностью, поэтому нам нужно посчитать значение функции Лебега в точке $x = 1.5$:

$$E_2 \leq \delta \left(\left| \frac{(1.5-1)(1.5-2)}{(0-1)(0-2)} \right| + \left| \frac{(1.5-0)(1.5-2)}{(1-0)(1-2)} \right| + \left| \frac{(1.5-0)(1.5-1)}{(2-0)(2-1)} \right| \right) =$$
$$= 5 \cdot 10^{-3} \cdot 1.25 = 6.25 \cdot 10^{-3}.$$

Итак, суммарная ошибка: $E \leq 4.6 \cdot 10^{-3} + 6.25 \cdot 10^{-3} = 1.085 \cdot 10^{-2}$. ■

5.6. Полиномиальная интерполяция на сетке Чебышева

Сетка Чебышева – это неравномерная сетка, построенная по нулям полинома Чебышева. На отрезке $[a, b]$ нули полинома Чебышева определяются по формуле:

$$x_k = \frac{a+b}{2} + \frac{b-a}{2} \cos \frac{2k-1}{2p} \pi, k = \overline{1, p}\,, \tag{19}$$

где p – число узлов сетки (степень полинома). Использование сетки Чебышева в задаче полиномиальной интерполяции позволяет минимизировать остаточный член (18), что повышает точность в целом. Для того, чтобы минимизировать остаточный член, необходимо минимизировать максимум модуля полинома $\prod_{j=0}^{N} (x - x_j)$. Ясно, что сеточные узлы – нули этого полинома. Существует

теорема, которая показывает, что среди всех таких полиномов наименьшее уклонение от нуля будет для полинома Чебышева. Можно показать, что уклонение полинома Чебышева степени p на отрезке $[a, b]$ будет

$$r = \left(\frac{b-a}{2}\right)^p \frac{1}{2^{p-1}}. \tag{20}$$

Замечание: отметим, что выражение (20) справедливо на всем отрезке $[a, b]$, несмотря на то, что первый и последний узел сетки Чебышева не совпадают с концами отрезка.

Задача 17 (ошибка интерполяции на сетке Чебышева). *Построить сетку из трех узлов, минимизирующую оценку остаточного члена полиномиальной интерполяции на отрезке* $[-2, 0]$ *и оценить ошибку интерполяции на этом отрезке. Третья производная интерполируемой функции на отрезке ограничена.*

Решение: сетка, минимизирующая оценку остаточного члена полиномиальной интерполяции – сетка Чебышева. Тогда воспользуемся формулой (19) и получим сетку $\{-\sqrt{3}/2-1, -1, \sqrt{3}/2-1\}$. Далее предположим, что третья производная интерполируемой функции на отрезке $[-2, 0]$ ограничена некоторой константой M_3. Тогда, воспользовавшись выражением (20), получим

$$\epsilon \leq \frac{M_3}{6} \cdot \frac{1}{2^{3-1}} = \frac{M_3}{24}. \blacksquare$$

5.7. Интерполяция сплайнами

Пусть на $[a, b]$ задана сетка $\{t_n\}_{n=0}^N$. *Сплайн* $S_m(t)$ – это определенная на $[a, b]$ функция, имеющая k непрерывных производных и являющаяся на каждом интервале (t_{n-1}, t_n) многочленом степени $\leq m$.

Дефект сплайна $d = m - k$, где m – степень сплайна, k – показатель гладкости.

Момент сплайна – значение второй производной функции $S(t)$ в узле сетки.

Кубическим сплайном дефекта 1, интерполирующим функцию $f(t)$, называется функция $S(t)$, удовлетворяющая следующим условиям:

1. $S(t_n) = f(t_n)$;
2. $S(t) \in C^2[a, b]$;
3. на любом $[t_n, t_{n+1}]$ $S(t)$ является кубическим многочленом;
4. на краях $[a, b]$ задано одно из следующих краевых условий:
   - $S'(a) = f'(a), S'(b) = f'(b)$,
   - $S''(a) = f''(a), S''(b) = f''(b)$ или $S''(a) = S''(b) = 0$,
   - $S(a) = S(b), S'(a) = S'(b)$ – условие периодичности.

Интерполяционный кубический сплайн $S(t)$, удовлетворяющий 1–3 и одному из условий 4 существует и единственен. Когда говорят, что надо провести сплайн-интерполяцию, понимают обычно интерполяцию кубическим сплайном дефекта 1. Такой сплайн интерполирует гладкую функцию с четвертым порядком точности, $O(\tau^4)$, где τ – шаг сетки. Также дифференцируя сплайн-интерполянт, можно получить интерполянты для первой и второй производной с точностью $O(\tau^3)$ и $O(\tau^2)$ соответственно.

Замечание: сплайн-интерполяция позволяет строить интерполянт по неограниченному количеству узлов.

Замечание: кусочно-линейная интерполяция – тоже интерполяция сплайнами степени 1 и дефекта 1.

Задача 18 (сплайн-интерполяция). *По заданным значениям функции*

t	-2	0	1
$f(t)$	-1	1	1

с помощью сплайн-интерполяции найти значение t, при котором $f(t) = 0$, если $f''(-2) = 0, f''(1) = -6$.

Решение: **шаг первый** – найдем все моменты сплайна. Моменты на краях находим из краевых условий, $m_0 = 0$ и $m_2 = -6$. Моменты во внутренних узлах находим из условий сшивки. Для каждого внутреннего узла верно:

$$\frac{f_{n+1} - f_n}{\tau_n} - \frac{f_n - f_{n-1}}{\tau_{n-1}} = m_{n-1}\frac{\tau_{n-1}}{6} + m_n\frac{\tau_n + \tau_{n-1}}{3} + m_{n+1}\frac{\tau_n}{6}.$$

Здесь $\tau_n = t_{n+1} - t_n$. В нашем случае всего один внутренний момент:

$$\frac{f_2 - f_1}{\tau_1} - \frac{f_1 - f_0}{\tau_0} = m_0\frac{\tau_0}{6} + m_1\frac{\tau_0 + \tau_1}{3} + m_2\frac{\tau_1}{6}.$$

Решая уравнение, находим $m_1 = 0$.

Шаг второй – найдем вид сплайна на каждом отрезке $[t_n, t_{n+1}]$. Общая формула:

$$S^{(n)} = \frac{1}{6\tau_n}(m_n(t_{n+1} - t)^3 + m_{n+1}(t - t_n)^3) + \alpha_n(t_{n+1} - t) + \beta_n(t - t_n),$$

$$\alpha_n = \frac{f_n}{\tau_n} - \frac{m_n\tau_n}{6},$$

$$\beta_n = \frac{f_{n+1}}{\tau_n} - \frac{m_{n+1}\tau_n}{6}.$$

В задаче

$$\alpha_0 = \frac{f_0}{\tau_0} - \frac{m_0\tau_0}{6} = \frac{-1}{2},$$

$$\beta_0 = \frac{f_1}{\tau_0} - \frac{m_1\tau_0}{6} = \frac{1}{2}.$$

И так как $m_0 = m_1 = 0$, то

$$S^{(0)} = \frac{-1}{2}(0 - t) + \frac{1}{2}(t + 2) = t + 1.$$

Аналогично, $S^{(1)} = -t^2 + t + 1$. $S^{(1)} > 0$ на $[0, 1]$, поэтому из $S^{(0)} = t + 1 = 0$ находим ответ: $\mathbf{t} = -\mathbf{1}$ – точка, в которой $f(t) = 0$. ■

6. Численные методы решения нелинейных уравнений

Рассмотрим нелинейное уравнение в векторном виде (то есть, по сути, систему уравнений):

$$\boldsymbol{f}(\boldsymbol{x}) = \boldsymbol{0}. \tag{21}$$

Классический подход численного решения нелинейных уравнений – использовать итерационный метод, позволяющий по начальному приближению уточнять решение с любой заданной точностью. Рассмотрим итерационный подход на двух примерах: на методе простой итерации (МПИ) и методе Ньютона. Начнем с метода простой итерации.

6.1. Метод простой итерации

МПИ основан на преобразовании исходного уравнения к уравнению вида:

$$\boldsymbol{x} = \boldsymbol{\phi}(\boldsymbol{x}). \tag{22}$$

Введём важное определение: отображение $\boldsymbol{\phi}$ называется *сжимающим* на метрическом (мы будем рассматривать нормированное) пространстве, если для любых двух точек $\boldsymbol{x}, \boldsymbol{y}$ выполняется

$$\rho(\boldsymbol{\phi}(\boldsymbol{x}), \boldsymbol{\phi}(\boldsymbol{y})) \leq q\rho(\boldsymbol{x}, \boldsymbol{y}),\ 0 < q < 1. \tag{23}$$

МПИ называется метод, который может быть представлен в виде:

$$\boldsymbol{x}_{n+1} = \boldsymbol{\phi}(\boldsymbol{x}_n).$$

Здесь n – номер итерации. Если отображение $\boldsymbol{\phi}$ является сжимающим в полном пространстве, то уравнение $\boldsymbol{x} = \boldsymbol{\phi}(\boldsymbol{x})$ имеет единственный корень $\boldsymbol{x}^*$, и МПИ сходится с любого начального приближения $\boldsymbol{x}_0$, причем:

$$\begin{aligned} &\|\boldsymbol{x}_n - \boldsymbol{x}^*\| \leq q^n \|\boldsymbol{x}_0 - \boldsymbol{x}^*\|; \\ &\|\boldsymbol{x}_n - \boldsymbol{x}^*\| \leq \frac{q^n}{1-q} \|\boldsymbol{x}_1 - \boldsymbol{x}_0\|. \end{aligned} \tag{24}$$

В одномерном случае в качестве полного пространства выступает обычно $\mathbb{R}^1$ либо отрезок локализации корня.

Используя эту теорему можно оценить число итераций для достижения заданной точности ϵ. Для этого нужно прологарифмировать неравенства и выразить из них n:

$$\|\boldsymbol{x}_n - \boldsymbol{x}^*\| \le q^n \|\boldsymbol{x}_0 - \boldsymbol{x}^*\| \le \epsilon \Rightarrow n \ge \frac{\log(\epsilon / \|\boldsymbol{x}_0 - \boldsymbol{x}^*\|)}{\log(q)}. \qquad (25)$$

Аналогичную оценку можно получить для второго варианта.

В случае, когда ϕ гладкая в области локализации корня для сходимости итерационного процесса, достаточно выполнения следующего условие:

$$\left\| \begin{matrix} \frac{\partial \phi_0}{\partial x_0} & \cdots & \frac{\partial \phi_0}{\partial x_N} \\ \cdots & \cdots & \cdots \\ \frac{\partial \phi_N}{\partial x_0} & \cdots & \frac{\partial \phi_N}{\partial x_N} \end{matrix} \right\| \le q < 1, \qquad (26)$$

где q – оценка скорости сходимости. Когда решается не система, а скалярное нелинейное уравнение, условие имеет более простой вид:

$$|\phi'(x)| \le q < 1. \qquad (27)$$

Это условие следует из теоремы о среднем.

Задача 19 (МПИ для решения нелинейного уравнения). *Построить сходящийся МПИ для решения нелинейного уравнения* $x^3 + 3x^2 - 1 = 0$.

Решение: сначала необходимо локализовать корни. Данное уравнение имеет не более трех вещественных корней. Можно убедиться, что корни локализованы на отрезках $[-3, -2]$, $[-1, 0]$ и $[0, 1]$, проверяя значения функции на концах. Рассмотрим отрезок $[-3, -2]$. На нем элементарными преобразованиями можно представить нелинейное уравнение в виде $x = x^{-2} - 3$, тогда $\phi(x) = x^{-2} - 3$. Убедимся, что итерационный процесс с такой правой частью сходится в области локализации. Для этого проверим выполнение условия (27): $|\phi'(x)| = |-2x^{-3}| \le \frac{1}{4} < 1$. То есть для нахождения корня на отрезке $[-3, -2]$ можно использовать метод простой итерации: $x_{k+1} = x_k^{-2} - 3$. Аналогично можно показать, что для отрезков $[-1, 0]$ и $[0, 1]$ подходящими МПИ будут $x_{k+1} = -(x_k + 3)^{-1/2}$ и $x_{k+1} = (x_k + 3)^{-1/2}$ соответственно. ■

Замечание: универсальной процедуры локализации корней нет. Для скалярного уравнения корни можно локализовать, например, графически. Или вычисляя значения функции в некоторой окрестности, чтобы найти отрезки, где функция будет менять знак.

Универсального способа построения наилучшего МПИ нет. Это чистое творчество. Однако, можно предложить общий вид метода для

случая простого корня, т.е. когда $f'(x^*) \neq 0$. Предположим, что на отрезке локализации корня производная не меняет знак, например $f'(x) > 0$. Допустим также, что у нас есть оценки на значения производной $f'(x) \in [m, M]$. Тогда преобразуем уравнение $f(x) = 0$ к следующему виду:

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

$\phi'(x) = 1 - \tau f'(x)$. Для сходимости МПИ достаточно условия $\max|\phi'(x)| < 1$. Раскроем модуль:

$$1 - \tau f'(x) < 1 \Rightarrow \tau f'(x) > 0 \Rightarrow \tau > 0,$$
$$1 - \tau f'(x) > -1 \Rightarrow \tau f(x) < 2 \Rightarrow \tau < \frac{2}{f(x)}.$$

Следовательно, метод будет сходиться при $\tau \in (0, \frac{2}{M})$. Такой метод иногда называют *методом релаксации*, похожий метод для решения систем линейных уравнений называется *метод Ричардсона*.

Кроме того, можно подобрать оптимальное значение τ, чтобы величина q в оценке:

$$|\phi'(x)| = |1 - \tau f'(x)| \leq q,\ x \in [x_l, x_r] \tag{28}$$

была минимальной. Оптимальные значения:

$$\tau_O = \frac{2}{m+M},\ q_O = \frac{M-m}{M+m} = \frac{1 - m/M}{1 + m/M}. \tag{29}$$

Задача 20 (метод релаксации). *Оцените количество итераций для вычисления корня уравнения $e^x + x^3 = 0$ с точностью $\epsilon = 10^{-6}$.*

Решение: Итак, $f(x) = e^x + x^3$. Нетрудно графически определить, что уравнение имеет единственный корень, $f(-1) < 0, f(0) > 0 \Rightarrow x^* \in \in [-1, 0]$.

$f'(x) = e^x + 3x^2 > 0$, поэтому метод релаксации применим. Можно грубо оценить, что $f'(x) > e^{-1} + 0 > 1/3$, и $f'(x) < e^0 + 3 \cdot 1^2 = 4$. В обоих случаях мы взяли минимум/максимум каждого слагаемого. Значит $m = 1/3$, $M = 4$.

Выберем оптимальный шаг

$$\tau_O = \frac{2}{M+m} = \frac{6}{13}.$$

В этом случае

$$q = \frac{M-m}{M+m} = \frac{11}{13};$$

$$e_n \le q^n e_0 \le \epsilon \Rightarrow n \ge \frac{\log(\epsilon/e_0)}{\log(q)} = \frac{\log(10^{-6}/0.5)}{\log(11/13)} = 78.55\,.$$

Следовательно, достаточно 79 итераций. ■

Задача 21 (порядок сходимости МПИ). *Определить порядок сходимости итерационного метода при вычислении корня* $x^* = \sqrt{a}(a > 0)$ *по формуле*:

$$x_{k+1} = x_k - \frac{11x_k^4 - 4x_k^2 a + a^2}{16x_k^5}.$$

Решение: чтобы проверить порядок сходимости МПИ, можно проверить следующее условие. Если

$$\phi'(x^*) = \phi''(x^*) = \dots \phi^{(n-1)}(x^*) = 0, \phi^{(n)}(x^*) \ne 0,$$

тогда

$$|x_{k+1} - x^*| \le C|x_k - x^*|^n.$$

В данной задаче несложно убедиться, что $\phi'(\sqrt{a}) = \phi''(\sqrt{a}) = \phi'''(\sqrt{a}) = 0$, а $\phi^{(4)}(\sqrt{a}) \ne 0$. То есть у данного метода четвертый порядок сходимости. ■

Задача 22 (МПИ для решения системы нелинейных уравнений). *Предложить МПИ для решения системы*

$$\begin{cases} y = \ln x, \\ x^2 + 3y^2 = 1 \end{cases}$$

и проверить выполнение достаточного условия сходимости к корню $x^* \approx 0.64, y^* \approx -0.44$.

Решение: корень в этой задаче уже локализован. Для сходимости МПИ достаточно, чтобы выполнялось условие (26). Глядя на систему уравнений, можно заметить, что правая часть в первом уравнении зависит только от одной переменной. Также в окрестности корня можно преобразовать второе уравнение так, чтобы его правая часть тоже зависела только от одной переменной. В этом случае в матрице Якоби функции ϕ два из четырех элементов будут нулевыми, что упростит вычисление нормы. Представим исходную систему в окрестности корня в виде:

$$\begin{cases} x = e^y = \phi_0(y), \\ y = -\frac{1}{\sqrt{3}}\sqrt{1 - x^2} = \phi_1(x). \end{cases}$$

Для такой системы матрица Якоби правой части будет иметь вид:

$$\frac{d\phi}{d\bar{x}} = \begin{pmatrix} 0 & \frac{\partial \phi_0}{\partial y} \\ \frac{\partial \phi_1}{\partial x} & 0 \end{pmatrix}.$$

и в первой норме достаточное условие сходимости сводится к проверке условия

$$\max\left(\left|\frac{\partial \phi_0}{\partial y}(y^*)\right|, \left|\frac{\partial \phi_1}{\partial x}(x^*)\right|\right) \le q < 1.$$

Здесь проверяем условие сходимости непосредственно в корне, считая рассматриваемую окрестность корня достаточно малой. Вычисляя $\frac{\partial \phi_0}{\partial y}(y^*) \approx 0.64$, $\frac{\partial \phi_1}{\partial x}(x^*) \approx 0.48$, убеждаемся, что условие сходимости в малой окрестности корня выполнено. ■

6.2. Метод Ньютона

Рассмотрим скалярное нелинейное уравнение

$$f(x) = 0 \tag{30}$$

на отрезке $[a, b]$. Сформулируем **теорему** (достаточное условие) о сходимости метода Ньютона: если $f(a)f(b) < 0$, причем f' и f'' непрерывны и знакопостоянны на $[a, b]$, то по начальному приближению $x_0 \in [a, b]$ такому, что

$$f(x_0)f''(x_0) > 0, \tag{31}$$

можно вычислить единственный корень уравнения (30) с любой точностью с помощью итерационного метода Ньютона:

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

Метод Ньютона имеет второй порядок сходимости:

$$|x_{n+1} - x^*| \le \alpha |x_n - x^*|^2.$$

Задача 23 (метод Ньютона для решения нелинейного уравнения). *У функции* $f(x) = 32x^3 + 20x^2 - 11x + 3 = 0$ *существует единственный вещественный корень на отрезке* $[-1.625, -0.08]$. *Указать нулевое приближение метода Ньютона.*

Решение: для решения задачи воспользуемся теоремой о сходимости метода Ньютона. Сначала проверим непрерывность и знакопостоянность первой и второй производных функции f: $f'(x) = 96x^2 + 40x - 11$,

$f''(x) = 192x+40$. Непрерывность очевидна. Для определения областей знакопостоянства найдем нули первой и второй производных. Решая квадратное уравнение, находим, что нули первой производной $x_1 > 0$ и $x_2 \approx -0.606$. x_1 лежит вне рассматриваемого отрезка. Единственный ноль второй производной $x_3 \approx -0.208$. Тогда области знакопостоянства первой и второй производных на отрезке $[-1.625, -0.08]$ определяются нулями x_2 и x_3: $[-1.625, -0.606)$, $(-0.606, -0.208)$, $(-0.208, -0.08]$. Среди этих областей разные знаки функции f на концах только на $[-1.625, -0.606)$. Причем $f(-1.625) < 0$ и $f(-0.606) > 0$. Будем далее рассматривать ее как область локализации корня. На этой области $f''(x) < 0$ всюду. Тогда, чтобы удовлетворить условию сходимости (31), выберем в качестве начального приближения левый край области локализации $x_0 = -1.625$. ■

Метод Ньютона для решения системы нелинейных уравнений (21) будет иметь вид:

$$\bar{x}_{n+1} = \bar{x}_n - \mathbf{f}_x^{-1}(\bar{x}_n)\mathbf{f}(\bar{x}_n), \tag{32}$$

где $\mathbf{f}_x^{-1}$ – матрица, обратная к матрице Якоби для функции $\mathbf{f}$.

Задача 24 (метод Ньютона для решения СНАУ). *Написать формулу Ньютона для системы*

$$\begin{cases} x^2 + y^2 = 1, \\ e^x - e^y = 1. \end{cases}$$

Решение: $\mathbf{f} = (f_1, f_2)^T$, где $f_1(x, y) = x^2 + y^2 - 1$, $f_2(x, y) = e^x - e^y - 1$. Тогда

$$\mathbf{f}_x = \begin{pmatrix} 2x & 2y \\ e^x & -e^y \end{pmatrix},$$

$$\mathbf{f}_x^{-1} = \frac{1}{2(xe^y + ye^x)} \begin{pmatrix} e^y & 2y \\ e^x & -2x \end{pmatrix}.$$

Подставляя $\mathbf{f}$ и $\mathbf{f}_x^{-1}$ в (32), после матричного умножения и расставления итерационных индексов, получим:

$$x_{n+1} = x_n - \frac{e^{y_n}(x_n^2 + y_n^2 - 1) + 2y_n(e^{x_n} - e^{y_n} - 1)}{2(x_n e^{y_n} + y_n e^{x_n})},$$

$$y_{n+1} = y_n - \frac{e^{x_n}(x_n^2 + y_n^2 - 1) - 2x_n(e^{x_n} - e^{y_n} - 1)}{2(x_n e^{y_n} + y_n e^{x_n})}. \blacksquare$$

7. Численное интегрирование

7.1. Квадратурные формулы Ньютона–Котеса

Пусть на отрезке $[a,b]$ задана сетка $\{t_n\}_{n=0}^N$ и в узлах сетке определена сеточная функция $\{f_n\}_{n=0}^N$, $f_n = f(t_n)$, которая является проекцией некоторой функции $f(t)$ на сетку. В задаче численного интегрирования необходимо по этим данным вычислить интеграл $\tilde{I} = \int\limits_a^b f(t)dt$ с некоторой заданной точностью ϵ. Далее без потери общности будем считать, что сетка равномерная с постоянным шагом τ.

Замечание: в формулах ниже будут использованы также значения $f_{n+\frac{1}{2}}$ – значения сеточной функции, взятые в полуцелом узле $t_{n+\frac{1}{2}} = t_n + \frac{\tau}{2}$. Возникает вопрос: откуда взять значение в полуцелом узле, если сеточная функция задана только в целых узлах? На самом деле использование полуцелых узлов лишь вопрос обозначений. Ничто не мешает в сетке только с целыми узлами каждый второй узел назвать полуцелым. Тогда число целых узлов уменьшится вдове, а сеточный шаг увеличится вдвое.

Рассмотрим три основных метода вычисления интеграла, объединенных общим названием *формулы Ньютона–Котеса*:

1. формула трапеций:
$$I = \sum_{n=0}^{N-1} \frac{1}{2}(f_n + f_{n+1})\tau, \text{ точность метода } \epsilon \le \frac{\max\limits_{[a,b]} |f''(\xi)|}{12}\tau^2(b-a);$$

2. формула прямоугольников:
$$I = \sum_{n=0}^{N-1} f_{n+\frac{1}{2}}\tau, \text{ точность метода } \epsilon \le \frac{\max\limits_{[a,b]} |f''(\xi)|}{24}\tau^2(b-a);$$

3. формула Симпсона:
$$I = \sum_{n=0}^{N-1} (f_n + 4f_{n+\frac{1}{2}} + f_{n+1})\frac{\tau}{6}, \text{ точность метода } \epsilon \le \frac{\max\limits_{[a,b]} |f^{(4)}(\xi)|}{2880}\tau^4(b-a).$$

Ключевая идея при выводе формул Ньютона–Котеса – проинтерполировать функцию на каждом шаге, а затем проинтегрировать проинтерполированную функцию. В частности, формула трапеций получается при полиномиальной интерполяции по двум узлам на каждом шаге интегрирования (интерполянт – линейная функция), формула Симпсона – по трем узлам (интерполянт – парабола). Таким образом можно получать формулы Ньютона–Котеса и более высокой точности, но с ростом числа узлов равномерной сетки, будут возникать проблемы.

Задача 25 (погрешность интегрирования). *Задана сеточная функция*

x	0	1	2	3	4
$f(x)$	4	3	3.2	2.5	2

Известно, что сеточная функция является проекцией некоторой бесконечно дифференцируемой функции f. Найти интеграл этой функции с использованием формулы трапеции. Оценить погрешность.

Решение: по формуле трапеции искомый интеграл $I_1 = 0.5 \cdot 4 + 3 + +3.2 + 2.5 + 0.5 \cdot 2 = 11.7$. Поскольку ничего неизвестно про вторую производную функции f, то провести оценку погрешности по формуле $\epsilon \leq \frac{\max\limits_{[a,b]} |f''(\xi)|}{12} \tau^2 (b-a)$ проблематично. Можно, конечно, попробовать вычислить производную численно, но в этом случае при вычислении производной тоже возникнет погрешность, и в оценке погрешности интеграла будет использовано значение с погрешностью, что увеличивает неопределенность. Воспользуемся для оценки погрешности интегрирования *правилом Рунге*. По правилу Рунге для погрешности справедливо:

$$\epsilon = |\tilde{I} - I_1| \approx \frac{|I_1 - I_2|}{2^p - 1}. \tag{33}$$

Здесь $\tilde{I}$ – точное значение интеграла, I_1 – интеграл на сетке с шагом τ, I_2 – интеграл на сетке с шагом 2τ, p – порядок точности метода. Возвращаясь к задаче, вычислим интеграл I_2 по трем узлам с шагом 2. $I_2 = 2 \cdot (0.5 \cdot 4 + 3.2 + 0.5 \cdot 2) = 12.4$. Порядок метода трапеций $p = 2$. Тогда погрешность $\epsilon = \frac{|11.7 - 12.4|}{2^2 - 1} \approx 0.24$. ■

Замечание: правило Рунге выводится из соображений, что $\tilde{I} = I_1 + +c \cdot h^p$, $\tilde{I} = I_2 + c \cdot (2h)^p$. Исключая константу c, которая определяется соответствующей производной в члене ошибки, получается правило. Здесь делается предположение, что c в обоих выражениях одно и то же, что, вообще говоря, не так, но различие будет тем меньше, чем h будет меньше.

7.2. Вычисление несобственных интегралов по формулам Ньютона–Котеса

При вычислении несобственных интегралов формулы Ньютона–Котеса нельзя использовать напрямую из-за наличия особенностей. Можно выделить четыре способа, позволяющих справиться с этой проблемой:

1. замена переменной;

2. интегрирование по частям;

3. представление в виде суммы собственного интеграла, вычисляемого численно, и несобственного, вычисляемого аналитически;

4. представление в виде суммы собственного интеграла, вычисляемого численно, и несобственного, оцениваемого как часть ошибки вычислений.

Рассмотрим эти подходы на примере задачи.

Задача 26 (вычисление несобственного интеграла). *С помощью формулы трапеций вычислить интеграл*

$$I = \int\limits_0^1 \frac{\cos x}{\sqrt{x}} dx$$

с точностью ϵ.

Решение: из условия задачи видно, что интеграл имеет особенность в нуле. Поэтому сразу задать сетку на $[0, 1]$ и вычислить интеграл по формуле трапеций не получится. Попробуем сначала справиться с этой проблемой с помощью *замены переменной*: используем замену $x = t^2$. Тогда $dx = 2tdt = 2\sqrt{x}dt$ и, после подстановки, $I = 2\int\limits_0^1 \cos t^2 dt$. Этот интеграл собственный и вычисляется по формуле трапеций с любой заданной точностью.

Теперь попробуем *проинтегрировать по частям*. $I = 2\sqrt{x}\cos x|_0^1 + 2\int\limits_0^1 \sqrt{x}\sin x dx = 2\cos 1 + 2\int\limits_0^1 \sqrt{x}\sin x dx$. Интеграл, получившийся после интегрирования по частям, собственный, который формально можно вычислить численно по формуле трапеций. Но при проведении оценки погрешности здесь возникает проблема, так как вторая производная подынтегральной функции в нуле равна ∞.Чтобы преодолеть эту проблему, надо проинтегрировать функцию по частям еще два раза: $I = 2\sqrt{x}\cos x|_0^1 + 2\int\limits_0^1 \sqrt{x}\sin x dx = 2\cos 1 + \frac{4}{3}x^{\frac{3}{2}}\sin x|_0^1 - -\frac{4}{3}\int\limits_0^1 x^{\frac{3}{2}}\cos x dx = 2\cos 1 + \frac{4}{3}\sin 1 - \frac{8}{15}\cos 1 - \frac{8}{15}\int\limits_0^1 x^{\frac{5}{2}}\sin x dx$. Теперь проблем со второй производной и с оценкой погрешности нет.

Следующий способ – *представить интеграл в виде суммы собственного интеграла, вычисляемого численно, и несобственного, вычисляемого аналитически*: $I = \int\limits_0^1 \frac{1}{\sqrt{x}}dx + \int\limits_0^1 \frac{\cos x - 1}{\sqrt{x}}dx$. Первый интеграл в сумме считается аналитически и равен 2. Второй интеграл без особенности и считается по формуле трапеций. По правилу Лопиталя можно показать, что предел подынтегральной функции в нуле равен нулю: $\lim\limits_{x\to 0} \frac{\cos x - 1}{\sqrt{x}} = \lim\limits_{x\to 0} \frac{(\cos x - 1)'}{(\sqrt{x})'} = \lim\limits_{x\to 0} -2\sqrt{x}\sin x = 0$.

Последний рассматриваемый способ – *представить интеграл в виде суммы собственного интеграла, вычисляемого численно, и несобственного, оцениваемого как часть ошибки вычислений.* Разобьем исходный интеграл на два: $I = I_1 + I_2$, где $I_1 = \int\limits_0^\delta \frac{\cos x}{\sqrt{x}}dx$, а $I_2 = \int\limits_\delta^1 \frac{\cos x}{\sqrt{x}}dx$. В интеграле I_2 особенности нет, так как нижний предел сдвинут от нуля. Он считается численно. А интеграл I_1 не будем считать, лишь оценим его и включим эту оценку в ошибку вычислений. Определимся, как выбирать δ. По условию задачи итоговая точность вычислений должна быть ϵ. Тогда вычислим интеграл I_2 с точностью $\epsilon/2$, а интеграл I_1 оценим как $\epsilon/2$. Оценка интеграла I_1 даст требуемое значение δ. Оценку проведем с помощью разложения в ряд Тейлора подынтегральной функции в окрестности нуля: $I_1 = \int\limits_0^\delta \frac{1 - \frac{x^2}{2} + \frac{x^4}{24} + O(x^6)}{\sqrt{x}}dx =$ $= 2\sqrt{\delta} - \frac{1}{5}\delta^{\frac{5}{2}} + \frac{1}{6}\delta^{\frac{9}{2}} + O(\delta^{\frac{11}{2}}) \le \frac{\epsilon}{2}$. Выбираем δ, чтобы удовлетворить условию задачи. ■

7.3. Метод Гаусса

Рассмотрим еще один метод интегрирования для вычисления интегралов вида

$$\int\limits_a^b \omega(x)f(x)dx.$$

Здесь $f(x)$ – некоторая заданная функция. $\omega(x)$ – весовая функция – некоторая функция специального вида, например, $\omega(x) = 1; e^{-x}; e^{-x^2}$. Ее использование обусловлено пределами интегрирования. В частности, если пределы конечны, то в качестве весовой функции можно использовать $\omega(x) = 1$. В дальнейшем на примере задачи будет более ясно ее назначение. Метод Гаусса строится на следующем факте: если $f(x) = P_m(x)$ – полином степени $m \le 2n-1$, то существуют такие веса

$\{A_i\}_{i=1}^n$ и узлы $\{x_i\}_{i=1}^n$, что точно выполняется следующее равенство:

$$\int_a^b \omega(x) P_m(x) dx = \sum_{i=1}^n A_i P_m(x_i). \tag{34}$$

То есть чтобы построить квадратурную формулу Гаусса для точного вычисления интеграла произвольного полинома степени не выше $2n-1$, необходимо найти n узлов и n весов.

Замечание: необходимо еще раз подчеркнуть, что для любого полинома степени не выше $2n-1$ набор узлов и весов в формуле (34) будет один и тот же.

Всего $2n$ неизвестных. Необходимо составить $2n$ уравнений для определения этих неизвестных. Рассмотрим базисный набор одночленов вида $\{1, x, x^2, \ldots, x^{2n-1}\}$ – всего $2n$ одночленов. Для всех этих одночленов можно составить $2n$ уравнений, в согласии с формулой (34), для определения $2n$ неизвестных значений узлов и весов:

$$\int_a^b \omega(x) x^j dx = \sum_{i=1}^n A_i x_i^j, j = \overline{0, 2n-1}. \tag{35}$$

Предполагается, что в этой формуле интеграл вычисляется аналитически. В итоге имеем нелинейную систему уравнений для определения весов и узлов. Далее можно перейти от ограничения на подынтегральную функцию $f(x)$ полиномом степени не выше $2n-1$ к произвольной функции из интерполяционных соображений. В этом случае равенство (34) уже не будет выполнятся точно, но точность будет достаточно высокой даже на малом количестве узлов.

Замечание: если пределы интеграла в формуле (35) конечны, то ясно, что интеграл вычисляется даже когда весовая функции равна 1. То есть квадратурная формула Гаусса строится для произвольной подынтегральной функции. Для вычислений в качестве весовой функции можно выделять какую-то специфическую часть функций из рассматриваемого класса. Например, можно выделить быстроосциллирующую часть для повышения точности вычислений, или быстрозатухающую для вычислений в неограниченных пределах.

Задача 27 (вычисления интегралов с бесконечным пределом). *Построить квадратурную формулу Гаусса для вычисления интеграла вида*

$$\int_0^\infty e^{-x} f(x) dx$$

по двум узлам.

Решение: весовая функция здесь $\omega(x) = e^{-x}$. Предлагается построить квадратурную формулу вида $\int\limits_0^\infty e^{-x} f(x)dx = A_1 f(x_1) + A_2 f(x_2)$. Необходимо найти четыре неизвестные: веса A_1, A_2 и узлы x_1, x_2. Для этого по базису $\{1, x, x^2, x^3\}$ составим систему нелинейных уравнений:

$$\begin{cases} \int\limits_0^\infty e^{-x}dx = A_1 + A_2 = 1, \\ \int\limits_0^\infty e^{-x}xdx = A_1x_1 + A_2x_2 = 1, \\ \int\limits_0^\infty e^{-x}x^2dx = A_1x_1^2 + A_2x_2^2 = 2, \\ \int\limits_0^\infty e^{-x}x^3dx = A_1x_1^3 + A_2x_2^3 = 6. \end{cases}$$

Решая систему, получим итоговую формулу

$$\int\limits_0^\infty e^{-x} f(x)dx = \frac{1}{2\sqrt{2}} \left[(\sqrt{2}+1) f(2-\sqrt{2}) + (\sqrt{2}-1) f(2+\sqrt{2}) \right]. \blacksquare$$

Замечание: аналогичным образом можно построить квадратуру для вычисления собственного интеграла быстроосциллирующей функции

$$\int\limits_{-\pi}^{\pi} e^x \sin 400x dx.$$

Точное значение интеграла -0.1154845, а квадратура Гаусса всего по двум узлам дает значение -0.1154855. То есть разница лишь в шестом знаке! Ясно, что с использованием квадратурных формул Ньютона–Котеса такой точности для быстроосциллирующей функции добиться гораздо сложнее.

8. Численные методы решения задачи Коши для ОДУ

Общий вид задачи Коши для системы ОДУ:

$$\mathbf{u}' = \mathbf{f}(t, \mathbf{u}), t \in [t_0, T],$$
$$\mathbf{u}(t_0) = \alpha.$$

Далее, все выкладки приведены для скалярного случая $u' = f(t, u)$, что никак не ограничивает общности. Положим $t_0 = 0$, введём сетку на отрезке $[0, T]$: $\{t_n = nh\}_{n=0}^{N}$, $h = T/N$. h – сеточный шаг. Будем обозначать u_n – приближённое значение функции u в узле t_n, $u(t_n)$ – значение точного решения в узле t_n.

Итак, имея начальное значение u_0 мы хотим вычислить $u_1, \ldots, u_N$ такие, что $u_n \approx u(t_n)$. Простейший метод – *явный метод Эйлера* – получается, если заменить u' односторонней разностью:

$$\frac{u_{n+1} - u_n}{h} = f(t_n, u_n), n = 0, 1, \ldots, N-1, \tag{36}$$
$$u_0 = \alpha. \tag{37}$$

В общем случае уравнения метода образуют систему *нелинейных* уравнений, т.к. неизвестные значения u_n входят как аргументы нелинейной функции f. Однако в данном случае удобнее рассматривать соотношения (36) как рекуррентное соотношение, т.к. можно явно найти u_{n+1}, зная u_n:

$$u_{n+1} = u_n + hf(t_n, u_n). \tag{38}$$

Стартуя с начального значения u_0, мы можем последовательно найти u_1, u_2 и т.д. Здесь проявляется «эволюционная природа» задачи Коши – мы двигаемся в положительном направлении по t, при этом нам нужно знать только начальное значение. В противоположность этому в краевой задаче значения на обеих концах интервала влияют на решение во всей области. Методы, в которых подобно методу Эйлера можно последовательно находить решение, двигаясь по времени, называются *маршевыми* (time-marching).

Если аппроксимировать производную u' в точке t_{n+1} разностью назад, получим *неявный метод Эйлера*:

$$\frac{u_{n+1} - u_n}{h} = f(t_{n+1}, u_{n+1}), n = 0, 1, \ldots, N-1, \tag{39}$$
$$u_0 = \alpha. \tag{40}$$

Или

$$u_{n+1} = u_n + hf(t_{n+1}, u_{n+1}).$$

Аналогично можно находить значения $u_1, u_2, \ldots$ последовательно, одно за другим, но теперь на каждом шаге нужно решать нелинейное уравнение (например, методом Ньютона), так как нет явной формулы, по которой можно найти значение u_{n+1} на каждом шаге.

Рассмотренные методы *одношаговые* – значение u_{n+1} зависит только от значения в предыдущем узле u_n и не зависит от других значений $u_{n-1}, u_{n-2}, \ldots$ Это соответствует дифференциальной постановке, т.к. точное решение в любой точке однозначно определяется *одним* значением (начальными данными). Однако существует целый класс *многошаговых* методов, в которых при вычислении u_{n+1} используются несколько предыдущих значений. Например, при замене u' центрально-разностной формулой, получается многошаговый метод (правило средней точки):

$$\frac{u_{n+1} - u_{n-1}}{2h} = f(t_n, u_n), \tag{41}$$

$$u_{n+1} = u_{n-1} + 2hf(t_n, u_n), n = 1, \ldots, N. \tag{42}$$

Важно отметить, что теперь помимо начального условия $u_0 = \alpha$, для того чтобы «запустить» счёт, необходимо как-то вычислить ещё одно значение u_1.

8.1. Порядок аппроксимации метода, невязка, локальная ошибка

Введём понятие *ошибки аппроксимации*. Ее еще называют *локальная ошибка* или *невязка*. Определим ее, как результат подстановки точного решения в разностные уравнения. *Порядок аппроксимации* – степень по шагу h, с которой входит старший член в ошибку аппроксимации.

Задача 28 (порядок аппроксимации явного метода Эйлера). *Покажите, что явный метод Эйлера имеет первый порядок аппроксимации.*

Решение: подставим в (36) точное решение. Тогда понятно, что равенство (36) уже не может выполняться точно. Можно сказать, что

$$\frac{u(t_{n+1}) - u(t_n)}{h} = f(t_n, u(t_n)) + r_n, \tag{43}$$

где r_n – локальная ошибка. Оценим r_n. Для этого разложим все входящие в (43) значения неизвестной функции u в ряд Тейлора относительно t_n. В данном случае необходимо разложить только $u(t_{n+1}) = = u(t_n) + hu'(t_n) + \frac{h^2}{2}u''(t_n) + O(h^3)$. После подстановки в (43) и преобразования, получим

$$u'(t_n) + \frac{h}{2}u''(t_n) + O(h^2) = f(t_n, u(t_n)) + r_n,$$

причем здесь $u'(t_n)$ сокращается с $f(t_n, u(t_n))$ по постановке задачи. Тогда получается, что невязка $r_n = \frac{h}{2}u''(t_n) + O(h^2)$. $u''(t_n)$ – некоторое фиксированное число, старший член невязки $\frac{h}{2}u''(t_n)$ имеет первый порядок по h. При стремлении h к нулю именно он будет давать основной вклад в ошибку аппроксимации. То есть $r_n = O(h)$ Поэтому говорят, что явный метод Эйлера имеет первый порядок аппроксимации. ■

Замечание: для явных методов, локальная ошибка – это ошибка, которую вносит метод на одном шаге по времени. Разумно ожидать, что если метод *устойчив* в каком-то смысле, то ошибка будет суммироваться от шага к шагу, но не будет слишком быстро расти (не заостряемся здесь на вопросе устойчивости и понимаем под ней пока именно написанное). Так как число шагов $N = \frac{T}{h}$, оценка ошибки явного метода Эйлера при вычислении u_{n+1} по u_n равна $r_n \cdot h = O(h) \cdot h$, тогда оценка полной ошибки в последнем узле будет $\frac{T}{h}O(h) \cdot h = O(h)$.

8.2. Методы Рунге–Кутты

Методы Эйлера – одношаговые методы первого порядка аппроксимации. Попробуем теперь построить одношаговый метод второго порядка аппроксимации. Следующий пример – демонстрация идеи построения методов высокого порядка. Запишем правило средней точки как одношаговый метод:

$$u_{n+1} = u_n + hf(t_n + \frac{h}{2}, u_{n+\frac{1}{2}}). \tag{44}$$

Этот метод имеет 2-й порядок аппроксимации, что гораздо лучше первого порядка метода Эйлера. Но какое значение взять для $u_{n+\frac{1}{2}}$? Ведь нам известно только u_n!

Попробуем немного изменить метод и вычислить $u_{n+\frac{1}{2}}$ приближённо с помощью явного метода Эйлера:

$$u_{n+\frac{1}{2}} = u_n + \frac{h}{2}f(t_n, u_n),$$
$$u_{n+1} = u_n + hf(t_n + \frac{h}{2}, u_{n+\frac{1}{2}}).$$

Немного перепишем эти расчетные формулы в другом виде:

$$k_1 = f(t_n, u_n), \tag{45}$$

$$k_2 = f(t_n + \frac{h}{2}, u_n + \frac{h}{2}k_1), \tag{46}$$

$$u_{n+1} = u_n + hk_2. \tag{47}$$

Может показаться странным, что для вычисления k_2 мы предлагаем сделать шаг методом Эйлера, который имеет первый порядок. Но из формулы видно, что k_2 умножается на h, и мы ожидаем, что метод «сохранит» второй порядок. Исследование аппроксимации метода подтверждает это предположение.

Все предыдущие «подгоночные» рассуждение служат только наводящими соображениями для того, чтобы записать некоторое семейство методов в общем виде. Обобщение этой «методики» даёт класс методов *Рунге–Кутты*. Пусть $s > 0$ – число стадий или этапов, a_{ij}, b_i, c_i – вещественные коэффициенты. Тогда метод

$$k_1 = f(t_n, u_n), \tag{48}$$

$$k_2 = f(t_n + c_2 h, u_n + ha_{21}k_1), \tag{49}$$

$$k_3 = f(t_n + c_3 h, u_n + h(a_{31}k_1 + a_{32}k_2)), \tag{50}$$

$$\dots \tag{51}$$

$$k_s = f(t_n + c_s h, u_n + h(a_{s1}k_1 + \ldots + a_{s,s-1}k_{s-1})), \tag{52}$$

$$u_{n+1} = u_n + h(b_1 k_1 + \ldots + b_s k_s) \tag{53}$$

называется s-стадийным *явным методом Рунге–Кутты*.

Коэффициенты метода принято записывать в виде таблицы, которую называют *таблицей Бутчера*.

Ниже приведены условия порядка аппроксимации для методов Рунге–Кутты (первые три порядка):

1. Первый порядок:

$$\sum_{j=1}^{s} a_{ij} = c_i, i = 1, \ldots, s, \tag{54}$$

$$\sum_{j=1}^{s} b_j = 1; \tag{55}$$

2. второй порядок (+ к предыдущим условиям):

$$\sum_{j=1}^{s} b_j c_j = \frac{1}{2}; \tag{56}$$

3. третий порядок (+ к предыдущим условиям):

$$\sum_{j=1}^{s} b_j c_j^2 = \frac{1}{3}, \tag{57}$$

$$\sum_{i=1}^{s}\sum_{j=1}^{s} b_i a_{ij} c_j = \frac{1}{6}. \tag{58}$$

Задача 29 (таблица Бутчера для метода Эйлера с пересчетом). *Выписать таблицу Бутчера для метода, заданного формулами* (45).

Решение: запишем коэффициенты $\{a_{ij}\}$ в виде матрицы, под которой снизу выпишем строкой коэффициенты b_j, а слева столбцом – коэффициенты c_i. Тогда таблица Бутчера примет вид:

$$\begin{array}{c|cc} 0 & 0 & 0 \\ 1/2 & 1/2 & 0 \\ \hline & 0 & 1 \end{array}$$

■

Замечание: стоит отметить, что для явных методов Рунге–Кутты таблица $\{a_{ij}\}$ всегда нижнетреугольная с нулями на главной диагонали.

Замечание: явный и неявный методы Эйлера – тоже методы Рунге–Кутты. Число стадий у них равно 1.

Задача 30 (расчетные формулы метода Рунге–Кутты). *Выписать расчетные формулы для решения ОДУ второго порядка*

$$\begin{cases} \frac{d^2y}{dx^2} + 2x\frac{dy}{dx} = 0, \\ y(0) = 0, y'(0) = 1 \end{cases}$$

методом Эйлера с пересчетом.

Решение: сначала сведем ОДУ второго порядка к системе ОДУ первого порядка. Для этого введем новые переменные $y = u$, $y' = v$. Тогда получим систему первого порядка:

$$\begin{cases} \dot{u} = v, \\ \dot{v} = -2xv, \\ u(0) = 0, v(0) = 1. \end{cases}$$

В векторном виде имеем ОДУ $\mathbf{u}' = \mathbf{f}(x, \mathbf{u})$, где $\mathbf{u} = (u, v)^T$, $\mathbf{f}(x, \mathbf{u}) = (v, -2xv)^T$. Выпишем расчетные формулы метода Эйлера $\mathbf{u}_{n+1} = \mathbf{u}_n + \frac{h}{2}(\mathbf{k}_1 + \mathbf{k}_2)$, где

$$\begin{cases} \mathbf{k}_1 = (k_{11}, k_{12})^T = \mathbf{f}(x_n, \mathbf{u}_n) = (v_n, -2x_n v_n)^T, \\ \mathbf{k}_2 = (k_{21}, k_{22})^T = \mathbf{f}(x_n + h, \mathbf{u}_n + h\mathbf{k}_1) = (v_n + hk_{12}, -2(x_n + h)(v_n + hk_{12}))^T. \end{cases}$$

9. Жёсткие задачи Коши для систем обыкновенных дифференциальных уравнений

Вспоминая способы построения методов высокого порядка на примере явных методов Рунге–Кутты, важно отметить следующее:

- используя Липшиц-непрерывность функции $f(x,u)$, можно доказать, что методы Р–К устойчивы, т.е. численное решение будет сходится к точному с порядком аппроксимации;
- методика позволяет построить метод *любого* порядка точности;
- явный метод Рунге–Кутты вычислительно реализуется в виде последовательного (s раз) вычисления функции правой части при разных значениях аргументов – это «дешёвые» операции.

Кажется, что указанные пункты позволяют закрыть тему численных методов для решения задачи Коши. Однако это не так.

Рассмотрим задачу Коши:

$$u'(t) = -\sin t, u(0) = 1, t \in [0,2] \tag{59}$$

с точным решением

$$u(t) = \cos t. \tag{60}$$

Применим для решения этой задачи явный метод Эйлера. Локальная ошибка (невязка):

$$r(t) = \frac{1}{2}h^2 u''(t) + O(h^3) = -\frac{1}{2}h^2 \cos t + O(h^3). \tag{61}$$

Функция $f(t) = -\sin t$ не зависит от u, можно получить такую оценку для глобальной ошибки:

$$|E| \le \frac{T}{h}\|r\|_\infty = h \max_{t\in[0,2]} |\cos t| = h. \tag{62}$$

Если мы хотим вычислить решение с точностью $|E| \le 10^{-3}$, нужно взять $h = 10^{-3}$, и мы получим нужное решение после $T/h = 2000$ шагов. Действительно, вычисления дают $u_{2000} = -0.415692$ с ошибкой $E^{2000} = u_{2000} - \cos(2) = 0.4548 \times 10^{-3}$.

Теперь изменим уравнение

$$u'(t) = \lambda(u - \cos t) - \sin t, u(0) = 1. \tag{63}$$

Точное решение этой задачи по-прежнему $u(t) = \cos t$. Так как невязка r^n зависит только от точного решения (и не зависит от уравнения), мы по-прежнему надеемся получить нужную точность, взяв шаг $h = 10^{-3}$. Возьмём $\lambda = -10$, получается значение $u_{2000} = -0.416163$ с ошибкой $E^{2000} = 0.161 \times 10^{-4}$.

Теперь возьмём $\lambda = -2100$. Точное решение не изменяется, локальная ошибка тоже. Но теперь, если мы выполним расчёт с $h = 10^{-3}$, получим $u_{2000} = -0.2453 \times 10^{77}$ с ошибкой величины 10^{77}. Решение ведёт себя «неустойчиво», ошибка растёт экспоненциально со временем.

Итак, мы знаем, что явный метод Эйлера устойчив, и, следовательно, численное решение сходится к точному. И, разумеется, с достаточно маленькими шагами мы получим хорошие результаты. В чём же дело?

Для данного линейного уравнения запишем, как меняется глобальная ошибка от шага к шагу:

$$E^{n+1} = (1 + h\lambda)E^n - r_n. \tag{64}$$

Это выражение проясняет причину экспоненциального роста ошибки: на каждом шаге предыдущая ошибка умножается на $(1 + h\lambda)$. Для случая $\lambda = -2100$, $h = 10^{-3}$ получается $1 + h\lambda = -1.1$, следовательно ошибка на шаге m увеличится в $(-1.1)^{n-m}$ раз по выполнении n шагов. $(-1.1)^{2000} \approx 10^{82}$, что соответствует полученному в расчёте значению. Когда $\lambda = -10$, $1 + h\lambda = 0.99$ и ошибка убывает. Благодаря этому и получился хороший результат.

Важно понять, что экспоненциальный рост ошибки не противоречит устойчивости метода (и сходимости).

9.1. A-устойчивость (Absolute stability)

Вся дальнейшая теория строится на основе модельного уравнения

$$u'(t) = \lambda u(t). \tag{65}$$

Где $\lambda \in \mathbb{C}$ – комплексная константа. Случай $Re(\lambda) > 0$ соответствует экспоненциально растущим решениям, т.е. неустойчивым; он рассматриваться не будет.

Представим одношаговый метод, применённый к уравнению (65), в виде

$$u_{n+1} = R(h\lambda)u_n. \tag{66}$$

Функция $R(z)$ называется *функцией устойчивости* данного метода. Функцию устойчивости можно истолковать как численное решение на первом шаге по времени задачи Коши для уравнения (65) с начальными данными $u_0 = 1$, $z = \lambda h$. Или более подробно: точное решение задачи Коши имеет вид $e^{\lambda t}$ для $\lambda \in \mathbb{R}, \lambda < 0$ – это экспоненциально убывающее решение. Численное решение: $R(z)^n$. Если $|R(z)| < 1$, то численное решение убывает и моделирует поведение точного. Если же $|R(z)| > 1$, численное решение экспоненциально растёт и вообще не приближает точное. Требование, чтобы численное решение было ограниченно $\forall t$ (при фиксированном шаге h!), приводит к следующим определениям:

Область $S = \{z \in \mathbb{C}, |R(z)| \leq 1\}$ называется *областью устойчивости* метода с функцией устойчивости $R(z)$. Если $\lambda h \in S$, решение (и ошибка) экспоненциально убывает со временем.

Как мы видели, условие $|R(z)| = |1 + h\lambda| \leq 1$ для явного метода Эйлера приводит к сильному ограничению на шаг по времени:

$$h \leq \frac{2}{|\lambda|}, \tag{67}$$

при больших значениях $|\lambda|$ метод становится непригодным для вычислений.

В связи с этим, существует потребность в методах, для которых не возникает таких жёстких ограничений на шаг. Естественно, лучше, если не возникает никаких ограничений. Эту простую идею выразил Далквист в 1963 году, когда ввёл следующее понятие: метод называется *A-устойчивым*, если при его применении к уравнению $u' = \lambda u$ $(Re(\lambda) < 0)$ отсутствуют ограничения на шаг, связанные с устойчивостью. Это определение распространяется и на многошаговые методы.

Для одношаговых методов можно сформулировать это определение в терминах функции устойчивости: метод, имеющий область устойчивости

$$S \supset \mathbb{C}^- = \{z, Re(z) \leq 0\} \tag{68}$$

(т.е. область устойчивости целиком содержит левую полуплоскость), называется A-устойчивым.

Напомним, как формулируется класс одношаговых неявных методов *Рунге–Кутты* в общем случае. Пусть $s > 0$ – число стадий или

этапов, a_{ij}, b_i, c_i – вещественные коэффициенты. Тогда метод

$$k_1 = f(t_n + c_1 h, u_n + h(a_{11}k_1 + \ldots + a_{1,s}k_s)), \tag{69}$$
$$k_2 = f(t_n + c_2 h, u_n + h(a_{21}k_1 + \ldots + a_{2,s}k_s)), \tag{70}$$
$$k_3 = f(t_n + c_3 h, u_n + h(a_{31}k_1 + \ldots + a_{3,s}k_s)), \tag{71}$$
$$\ldots \tag{72}$$
$$k_s = f(t_n + c_s h, u_n + h(a_{s1}k_1 + \ldots + a_{s,s}k_s)), \tag{73}$$
$$u_{n+1} = u_n + h(b_1 k_1 + \ldots + b_s k_s) \tag{74}$$

называется s-стадийным *неявным методом Рунге–Кутты.*

Следующие формулы показывают связь функции устойчивости метода Рунге–Кутты с его коэффициентами. Функция устойчивости неявного (в общем случае) метода Рунге–Кутты выражается через коэффициенты так:

$$R(z) = 1 + z\boldsymbol{b}^T(\boldsymbol{I} - z\boldsymbol{A})^{-1}\boldsymbol{e}, \tag{75}$$

где $\boldsymbol{A} = [a_{ij}], \boldsymbol{b} = [b_1, \ldots, b_s]^T, \boldsymbol{e} = [1, 1, \ldots, 1]^T, \boldsymbol{I}$ – единичная матрица. Или так

$$R(z) = \frac{det(\boldsymbol{I} - z\boldsymbol{A} + z\boldsymbol{e}b^T)}{det(\boldsymbol{I} - z\boldsymbol{A})}. \tag{76}$$

Задача 31 (функция и область устойчивости метода Рунге–Кутты). *Для решения задачи Коши для системы ОДУ*

$$\begin{cases} \dot{u} = -800u + 0.04v + 0.02w, \\ \dot{v} = -5v - 3w, \\ \dot{w} = v - w, \\ u(0) = 0, v(0) = 4, w(0) = 6. \end{cases}$$

используется метод Рунге–Кутты с таблицей Бутчера:

1/5	*1/5*	*0*
4/5	*3/5*	*1/5*
	1/2	*1/2*

Получите для него функцию и условие устойчивости. Найдите показатель жесткости.

Решение: функция устойчивости получается по одной из формул (75) или (76):

$$R(z) = \frac{1 + 0.6z + 0.14z^2}{1 - 0.4z + 0.04z^2}. \tag{77}$$

Поскольку правая часть системы ОДУ линейная с постоянными коэффициентам, то заменой переменных можно представить систему ОДУ как три независимых модельных уравнения вида (65) с собственными числами матрицы, составленной из коэффициентов правой части, в качестве множителей λ. Эти собственные числа будут $\lambda_1 = -2, \lambda_2 = -4$ и $\lambda_3 = -800$. Так как $z = \lambda h$, то и функцию устойчивости достаточно исследовать на действительной оси. Условие устойчивости определяется областью устойчивости $|R(z)| \leq 1$. После решения неравенства получается, что условие устойчивости $z(1+z/10) \leq 0$. Проверяя его для всех λ, получаем h, который удовлетворяет всем случаям: $h \in \left(0, \frac{10}{800}\right)$. Показатель жесткости – отношение максимального и минимального по модулю собственного числа $s = \frac{800}{2} = 400$. ■

9.2. L-устойчивость, монотонность

A-устойчивость гарантирует, что $\forall \lambda, Re(\lambda) < 0$ численное решение «затухает» с ростом времени (числа шагов). Однако может оказаться, что $|R(z)|$ близко к единице и затухание происходит медленно. Это частично объясняет потребность в следующем определении: метод называется *L-устойчивым*, если он A-устойчив и

$$\lim_{z \to \infty} R(z) = 0. \tag{78}$$

Полезно помнить, что для рациональной функции (какой и является функции устойчивости) верно (i – мнимая единица):

$$\lim_{z \to -\infty} R(z) = \lim_{z \to \infty} R(z) = \lim_{z = iy,\, y \to \infty} R(z)\,. \tag{79}$$

Ещё одно желательное свойство – монотонность. Точное решение тестовой задачи Коши $e^{\lambda t} > 0$ – монотонно убывающая функция, поэтому разумно потребовать этого от численного решения: метод называется *монотонным*, если $\forall y \in \mathbb{R}, y < 0$, выполняется $0 < R(y) < 1$.

Задача 32 (ОДУ второго порядка). *Для решения задачи Коши для ОДУ второго порядка*:

$$\begin{cases} y'' = -\frac{19}{4} y - 10y', \\ y(0) = -9, y'(0) = 0 \end{cases} \tag{80}$$

используется метод трапеции. Найти показатель жесткости задачи и исследовать на L-устойчивость.

Решение: сначала сведем ОДУ второго порядка к системе первого порядка:

$$\mathbf{u}' = -A\mathbf{u}, \mathbf{u}(0) = (-9, 0)^T.$$

Здесь $\mathbf{u} = (y_1, y_2)^T$, $y = y_1' = y_2$, а матрица

$$A = \begin{pmatrix} 0 & -1 \\ \frac{19}{4} & 10 \end{pmatrix}. \tag{81}$$

Собственные числа матрицы $\lambda_1 = -1/2$, $\lambda_2 = -19/2$. Тогда показатель жесткости $s = 19$. Разностное уравнение метода трапеции имеет вид

$$\frac{u_{n+1} - u_n}{h} = \frac{f_n + f_{n+1}}{2}. \tag{82}$$

Применяя его для модельного уравнения, получим функцию устойчивости $R(z) = \frac{2+z}{2-z}$. Несложно видеть, что область устойчивости метода – вся левая комплексная полуплоскость. Значит метод A-устойчивый. Но он не будет L-устойчивым, так как условие (78) не выполняется. ■

9.3. A-устойчивость многошаговых методов

Для исследования многошагового метода на A-устойчивость можно воспользоваться, например, теоремой, которая называется *второй барьер Далквиста*: любой A-устойчивый многошаговый метод должен иметь порядок $p \leq 2$.

Чтобы найти порядок многошагового метода вида

$$\alpha_k y_{l+k} + \alpha_{k-1} y_{l+k-1} + \ldots + \alpha_0 y_l = h(\beta_k f_{l+k} + \beta_{k-1} f_{l+k-1} + \ldots + \beta_0 f_l), \tag{83}$$

где $f_i = f(x_i, y_i), \alpha_k \neq 0, |\alpha_0| + |\beta_0| > 0$, можно воспользоваться условием порядка. Многошаговый метод имеет порядок p, если

$$\sum_{j=0}^{k} \alpha_j = 0, \sum_{j=0}^{k} \alpha_j j^q = q \sum_{j=0}^{k} \beta_j j^{q-1}, q = 1, \ldots, p. \tag{84}$$

Задача 33 (A-устойчивость многошагового метода). *Найти порядок метода и исследовать его на A-устойчивость:*

$$\frac{x_n + x_{n-1} - 2x_{n-2}}{3\tau} = \frac{1}{12} f_n + \frac{4}{6} f_{n-1} + \frac{3}{12} f_{n-2}. \tag{85}$$

Решение: воспользуемся условием порядка для многошагового метода и получим, что $p = 3$. По второму барьеру Далквиста метод не может быть A-устойчивым. ■

ЛИТЕРАТУРА

1. *Петров И.Б., Лобанов А.И.* Лекции по вычислительной математике. Интернет-Ун-т Информ. Технологий, БИНОМ. Лаб. знаний, 2006.
2. *LeVeque R.J.* Finite difference methods for ordinary and partial differential equations: steady-state and time-dependent problems. Society for Industrial and Applied Mathematics, 2007.
3. *Johnston N.* Numerical methods in engineering with matlab // Proceedings of the Institution of Mechanical Engineers. 2010. V. 224, N B7. P. 1158.
4. *Аристова Е.Н., Завьялова Н.А., Лобанов А.И.* Практические занятия по вычислительной математике. Ч. 1. Москва : МФТИ. 2014.
5. *Демченко В.В.* [*и др.*]. Упражнения и задачи контрольных работ по вычислительной математике. Ч. 1, Ч. 2. Москва : МФТИ, 2014.
6. *Gautschi W.* Numerical analysis. Springer Science & Business Media, 1997.
7. *Johansson R.* Numerical Python: A Practical Techniques Approach for Industry. Apress, 2015.