

Федеральное государственное бюджетное
образовательное учреждение высшего образования
«Московский государственный технический университет имени Н.Э. Баумана
(национальный исследовательский университет)»

М.П. Галанин, В.В. Лукин, О.В. Щерица

Методы вычислений Задачи алгебры и анализа

Учебное пособие



Москва

ИЗДАТЕЛЬСТВО

МГТУ им. Н. Э. Баумана

2022

УДК 519.6
ББК 22.193

Г15

Издание доступно в электронном виде по адресу
<https://bmstu.press/catalog/item/7709/>

Факультет «Фундаментальные науки»
Кафедра «Прикладная математика»

*Рекомендовано Научно-методическим советом
МГТУ им. Н.Э. Баумана в качестве учебного пособия*

*Рецензент
д-р физ.-мат. наук А.В. Колдоба*

Галанин, М. П.

Г15 Методы вычислений. Задачи алгебры и анализа :
учебное пособие / М. П. Галанин, В. В. Лукин, О. В. Щерица. —
Москва : Издательство МГТУ им. Н. Э. Баумана, 2022. —
376 с. : ил.

ISBN 978-5-7038-5770-0

Изложены основные численные методы и алгоритмы решения базовых математических задач, ориентированных на применение современной вычислительной техники и позволяющих эффективно проводить количественный анализ математических моделей широкого класса реальных природных, социальных и технических объектов. Приведены методы решения задач линейной алгебры, систем нелинейных алгебраических уравнений, методы интерполяции функций, численного интегрирования и дифференцирования, численные методы решения задач Коши для систем обыкновенных дифференциальных уравнений.

Для студентов старших курсов технических университетов, аспирантов и инженеров. Может быть полезно также преподавателям и научным работникам.

УДК 519.6
ББК 22.193

Учебное издание

**Галанин Михаил Павлович, Лукин Владимир Владимирович,
Щерица Ольга Владимировна**

**Методы вычислений
Задачи алгебры и анализа**

Оригинал-макет подготовлен в Издательстве МГТУ им. Н.Э. Баумана.
В оформлении использованы шрифты Студии Артемия Лебедева.

Подписано в печать 21.11.2022. Формат 60×90/16.

Усл. печ. л. 23,5. Тираж 157 экз. Изд. № 1007-2022 (4704).

Издательство МГТУ им. Н.Э. Баумана. 105005, г. Москва, улица 2-я Бауманская, д. 5, к. 1.
info@bmstu.press <https://bmstu.press>

Отпечатано в типографии МГТУ им. Н.Э. Баумана.
105005, г. Москва, улица 2-я Бауманская, д. 5, к. 1. baumanprint@gmail.com

© МГТУ им. Н.Э. Баумана, 2022
© Оформление. Издательство
МГТУ им. Н.Э. Баумана, 2022

ISBN 978-5-7038-5770-0

ПРЕДИСЛОВИЕ

Методы численного решения задач линейной алгебры и анализа лежат в основе большинства научных и инженерно-технических расчетов. Исследование математических моделей физических явлений и технических систем в XXI в. невозможно представить без применения вычислительной техники и эффективных численных алгоритмов.

Данное учебное пособие посвящено изложению ставших классическими базовых методов решения задач линейной алгебры, нахождения решения систем линейных и нелинейных алгебраических уравнений, методов интерполяции функций одного и нескольких пространственных переменных, методов определения собственных значений матрицы численного интегрирования и дифференцирования, а также методов решения обыкновенных дифференциальных уравнений. Структура пособия отражает блочно-модульное построение дисциплины «Методы вычислений», занимающей одно из ключевых мест в профессиональном цикле подготовки бакалавров по направлению «Прикладная математика». Курс читается в течение одного семестра, он включает по два часа лекций, практических занятий и лабораторных работ в неделю. Данная дисциплина основана на знании ряда обще-математических дисциплин, а именно курсов математического анализа, линейной алгебры, функционального анализа и других прикладных дисциплин.

Цель дисциплины — научить использовать при решении прикладных задач методы вычислительной математики, выбирать эффективные алгоритмы их программной реализации. Изложенные

в пособии методы лежат в основе многих современных прикладных программных комплексов, применяемых для математического моделирования физических явлений, технических систем и технологических процессов. Курс позволяет установить общую фундаментальную базу для дальнейшего изучения прикладных математических дисциплин, характерных для современных учебных планов технических университетов и связанных с численным решением задач прикладной математики.

В частности, логическим продолжением курса является дисциплина «Численные методы решения задач математической физики», предполагающая изучение методов конечных разностей, конечных элементов и других методов нахождения приближенного решения краевых задач. На методах, рассмотренных в пособии, базируется численное решение задач теории упругости и пластичности, газо- и гидродинамики, задач других разделов механики и физики. Общность и неразрывная связь перечисленных дисциплин позволяют рассматривать их как единое целое.

Курс основан на материале учебников и монографий по численным методам и на личном опыте авторов, как вычислительном, так и педагогическом. Значительная часть материала, использованная авторами, содержится в учебниках, приведенных в списке литературы. Среди них прежде всего следует отметить книгу М.П. Галанина и Е.Б. Савенкова «Методы численного анализа математических моделей», охватывающую как темы, рассматриваемые в данном издании, так и различные семейства численных методов решения задач математической физики. В настоящем издании многие вопросы изложены более подробно, с большей опорой на иллюстративный материал и примеры. С методической точки зрения материал пособия подобран так, чтобы читатель мог освоить программу дисциплины без

обращения к другим пособиям. Однако это не означает, что авторы исчерпали всю тематику методов вычислений. Список литературы, приведенный в конце пособия, включает основные издания по численным методам и дает представление об объеме материала, не вошедшего в пособие.

Авторы учились численным методам в основном по лекциям и книгам В.И. Агошкова, В.Б. Андреева, В.Я. Арсенина, Н.Н. Бахвалова, В.В. Воеводина, С.К. Годунова, А.В. Гулина, Ю.Н. Днестровского, Н.Н. Калиткина, Ю.Н. Карамзина, Д.П. Костомарова, Г.И. Марчука, А.Ф. Никифорова, Е.С. Николаева, Ю.П. Попова, Б.Л. Рождественского, В.С. Рябенького, А.А. Самарского, А.Г. Свешникова, А.Н. Тихонова, В.Б. Уварова, А.П. Фаворского, Р.П. Федоренко, Н.Н. Яненко. Эти ученые оказали значительное влияние на авторов, что нашло отражение в содержании и стиле изложения данного пособия.

В своей семинарской работе авторы используют известные сборники задач по численным методам, которые представлены в списке литературы. Текст содержит много примеров, призванных продемонстрировать возможности решения задач.

Область методов вычислений — хорошо разработанная и в то же время динамично развивающаяся часть современной математики. В связи с этим в список литературы наряду с учебными изданиями включены монографии, посвященные более узким темам, чем рассмотренные в пособии. В конце каждой главы приведены библиографические комментарии, дающие краткий обзор классических и современных работ по теме главы. Такие комментарии позволяют читателю проще ориентироваться в большом количестве литературы по вычислительным методам, получить более глубокие знания по рассматриваемой теме и могут быть полезны студентам при выполнении курсовых и выпускных работ.

Пособие содержит восемь глав, параграфы в которых имеют двойную нумерацию (например, 2.3 — третий параграф второй главы). Аналогично двойную нумерацию имеют рисунки, таблицы и теоремы (например, рис. 3.4 — четвертый рисунок в главе 3). Конец примеров отмечен значком •, начало и конец доказательства теорем, следствий — значками ◀ и ► соответственно.

Основные термины выделены в тексте **полужирным курсивом**. Светлым курсивом выделены термины, отнесенные к ключевым словам; для понимания излагаемого материала читатель должен знать значение этих терминов.

После предисловия помещен список основных обозначений, где наряду с их краткой расшифровкой указаны параграфы, в которых можно найти более подробное объяснение этих обозначений. Каждая глава снабжена вопросами и заданиями для самопроверки, позволяющими читателю оценить уровень усвоения изложенного материала.

Авторы выражают глубокую благодарность своим коллегам и ученикам по ИПМ им. М.В. Келдыша РАН и МГТУ им. Н.Э. Баумана за совместный труд, в результате которого появилась данная книга. Список коллег и учеников, к счастью, очень велик. Не имея возможности перечислить их всех, авторы особо благодарят Ю.П. Попова, Н.А. Тихонова и О.С. Мажорову, чьими учениками они являются.

ОСНОВНЫЕ ОБОЗНАЧЕНИЯ

- \forall — квантор всеобщности
 \exists — квантор существования
 $| \cdot |$ — абсолютное значение числа или модуль вектора
 $(\cdot)^T$ — транспонирование матрицы
 $[a, b], (a, b)$ — отрезок и интервал с концами в точках $a, b \in \mathbb{R}$
 a^*, y^* — приближенное значение величины a и приближенное значение функции $y^* = y(x^*)$ **1.1.2**
 $\|A\|$ — норма оператора либо матрицы A **1.2.3**
 $\|A\|_1$ — октаэдрическая норма матрицы $A_{n \times n}$ **1.2.6**
 $\|A\|_2$ — евклидова (сферическая, шаровая) норма матрицы $A_{n \times n}$ **1.2.6**
 $\|A\|_\infty$ — кубическая норма матрицы $A_{n \times n}$ **1.2.6**
 $\|A\|_M$ — максимальная норма матрицы A **1.2.6**
 $\|A\|_s$ — спектральная норма матрицы A **1.2.6**
 A^* — оператор, сопряженный оператору A **1.2.4**
 A^{-1} — оператор, обратный оператору A **1.2.3**
 $A: x \rightarrow y$ — оператор A отображает элемент x в элемент y
 $A: X \rightarrow Y$ — оператор A отображает множество X на (или в) множество Y
 $C[a, b]$ — пространство функций, непрерывных на отрезке $[a, b]$ **1.2.2**
 $D(A)$ — область определения оператора A **1.2.3**
 E — единичный оператор **1.2.3**
 \tilde{f} — интерполянт заданной функции f **4.1.1**

- $H_n(x)$ — алгебраический интерполяционный полином Эрмита степени n **4.2.4**
- \tilde{i} — мнимая единица **6.4**
- $\text{im } A$ — область значений оператора A **1.2.3**
- $\inf_{x \in X} f(x)$ — точная нижняя грань множества числовой оси $\{f(x), x \in X\} \subset \mathbb{R}$
- $\ker A$ — ядро оператора A **1.2.3**
- l_p — бесконечномерное пространство последовательностей вида $x = \{x_n\}$ **1.2.2**
- $L_2[a, b]$ — пространство функций, интегрируемых с квадратом в смысле Лебега **1.2.2**
- $L_n(x)$ — алгебраический интерполяционный полином степени n **4.2.2**
- $L_p[a, b]$ — пространство функций, удовлетворяющих условию $\int_a^b |x|^p dt < \infty$, где интеграл понимается в смысле Лебега, $1 < p < \infty$ **1.2.2**
- $n = \overline{1, N}$ — число $n \in \mathbb{N}$ принимает последовательно все значения из множества \mathbb{N} натуральных чисел от 1 до N включительно
- $P_n(x)$ — алгебраический полином степени n **4.2.1**
- $Q_n(x)$ — тригонометрический полином порядка n **4.3.5**
- \mathbb{R} — множество действительных чисел (числовая прямая)
- \mathbb{R}^n — пространство векторов вида $(x_1, x_2, \dots, x_n)^\top$ **1.2.2**
- $\sup_{x \in X} f(x)$ — точная верхняя грань множества числовой оси $\{f(x), x \in X\} \subset \mathbb{R}$
- $S_n(x)$ — интерполяционный сплайн степени n **4.4**
- $T_n(x)$ — полином Чебышёва степени n **4.3.1**
- $x \in X$ — элемент x принадлежит множеству X

$x_n \rightarrow x$ — последовательность элементов x_n сходится к x **1.2.1**

$\{x_n\}$, $\{x_n\}_0^\infty$ — последовательность элементов x_n **1.2.1, 2.6.1**

$\|x\|$, $\|x\|_H$ — норма элемента x линейного нормированного пространства H **1.2.1**

$\|x\|_D = (Dx, x)^{1/2}$ — энергетическая норма (D -норма) элемента x линейного нормированного пространства, соответствующая симметричному положительно определенному оператору D **1.2.4**

$\|x\|_1$ — октаэдрическая норма вектора $x \in \mathbb{R}^n$ **1.2.6**

$\|x\|_2$ — евклидова (шаровая) норма вектора $x \in \mathbb{R}^n$ **1.2.6**

$\|x\|_\infty$ — кубическая норма вектора $x \in \mathbb{R}^n$ **1.2.6**

(x, y) — скалярное произведение элементов x и y унитарного пространства **1.2.1**

$(x, y)_D = (Dx, y)$ — энергетическое скалярное произведение элементов x и y унитарного пространства, соответствующая симметричному положительно определенному оператору D **1.2.4**

$x \perp y$, $x \perp X$ — элемент x унитарного пространства ортогонален элементу y и элемент x унитарного пространства ортогонален подпространству X **1.2.1**

$X \subset Y$ — подмножество X включено в множество Y (Y включает X)

y_x — разностная производная сеточной функции вперед **6.7**

$y_{\bar{x}}$ — разностная производная сеточной функции назад **6.7**

y_x^o — центральная разностная производная сеточной функции **6.7**

$y_{\bar{x}x}$ — вторая разностная производная сеточной функции **6.7**

$W_2^1([a, b])$ — пространство Соболева **1.2.2**

δ_{ij} — символ Кронекера, функция двух целых переменных i и j , равная единице, если $i = j$, и нулю в противном случае **1.2.1**

$\delta(a^*)$ — относительная погрешность приближенного значения величины a **1.1.2**

$\Delta(a^*)$, $\Delta(y^*)$ — абсолютная погрешность приближенного значения величины a и линейная оценка абсолютной погрешности вычисления функции y **1.1.2**, **1.1.4**

$\Delta y_i = y_{i+1} - y_i$ — конечная разность **2.6.1**

$\Delta^n y_i = \Delta(\Delta^{n-1} y_i)$ — конечная разность n -го порядка **2.6.1**

$\rho(A)$ — спектральный радиус оператора A **1.2.3**

$\Phi[u]$ — значение функционала Φ на элементе u **8.3**

ВВЕДЕНИЕ

Математические расчеты лежат в основе большинства сфер человеческой деятельности: начиная с задач вычисления количества чего-либо, измерения длин, расстояний, площадей, объемов и заканчивая сложными вычислительными экспериментами по расчету гидродинамических течений, определению прочностных свойств конструкций, решением задач прогнозирования. По мере усложнения задач растет количество применяемых методов и подходов для их решения, часто новые задачи требуют создания новых алгоритмов. В то же время сложный расчет может быть редуцирован к набору относительно стандартных подзадач, таких как решение систем линейных алгебраических уравнений, интерполирование и аппроксимация функций, численное интегрирование и дифференцирование. Накопленный обширный опыт решения указанных задач привел к возникновению и развитию целого ряда вычислительных подходов, методов и приемов, каждый из них, конечно, не является универсальным, но оказывается наиболее эффективным в том классе задач, для которого он создавался.

Благодаря развитию вычислительной техники появились доступные пакеты программ для решения однотипных задач. Однако слепое доверие этим пакетам может привести к ошибкам, поскольку пользователь, не зная тонкостей работы алгоритма, может применить тот или иной стандартный и известный метод в задачах, особенности которых алгоритм метода не учитывает. Зачастую для переноса вычислительного метода с одной предметной области на другую требуются пересмотр

и видоизменение каждого из «кирпичиков» используемого алгоритма. Подобная работа требует знания и понимания основ построения численных методов и алгоритмов, поскольку, только зная их устройство, можно выполнить тонкую настройку как алгоритма, так и готового пакета программ для решения конкретных инженерных или научных задач.

Несмотря на высокий уровень развития современной вычислительной техники, возникают проблемы с реализацией даже хорошо изученных и исследованных алгоритмов: порой получение с их помощью результатов становится неэффективным. Например, такая ситуация может возникнуть при решении системы линейных алгебраических уравнений. Решение системы всегда может быть выписано в конечном и явном виде с помощью формулы Крамера. Однако численно решить эту задачу по формуле Крамера при размерностях системы, возникающих в инженерной вычислительной практике ($\sim 10^4 \dots 10^7$ уравнений), невозможно в силу значительных временных затрат. В итоге снова возникает проблема построения качественных вычислительных алгоритмов, которые за разумное время будут давать надежные результаты для решения тех или иных задач.

Еще одна особенность проведения расчетов с помощью вычислительной техники — представление чисел в ЭВМ. Для хранения чисел используется конечное количество разрядов. Числа можно представить либо с фиксированной, либо с плавающей запятой. Чаще всего они хранятся в виде числа с плавающей запятой. В памяти ЭВМ можно хранить лишь ограниченное количество разрядов мантиссы и порядка. И как следствие, в такой ЭВМ нельзя точно представить не только трансцендентные числа (бесконечную непериодическую десятичную дробь), но и рациональные числа (например, $1/3$), а также слишком большие или малые по абсолютному значению. В результате в ЭВМ вместо

точного действительного числа хранится его приближение в виде числа с плавающей запятой.

Эффективность работы вычислительного алгоритма также зависит от того, как в ЭВМ хранятся данные, как они упорядочены в ее памяти, как происходит обращение к ним. В частности, элементы матрицы в разных языках программирования в памяти ЭВМ упорядочены либо по строкам, либо по столбцам. В результате длительность работы одного и того же вычислительного метода, реализованного на разных языках программирования, может значительно различаться, даже если расчеты проводить на одной и той же ЭВМ. Яркий пример — задача перемножения матриц большого размера. С одной стороны, произведение матриц AB может быть рассмотрено как матрица, элементами которой являются всевозможные скалярные произведения вектор-строк матрицы A и вектор-столбцов матрицы B . С другой стороны, оно может быть представлено как сумма внешних произведений столбцов матрицы A и строк матрицы B . В зависимости от используемого языка программирования, способа хранения данных и архитектуры ЭВМ простое изменение порядка суммирования при вычислении элементов матрицы AB для матриц размерностью 1000×1000 может ускорить расчет в несколько раз.

Современная вычислительная техника предоставляет все больше возможностей для ускорения расчетов, однако их невозможно эффективно использовать без знания и понимания внутреннего устройства вычислительных алгоритмов, логики их функционирования, оценок скорости сходимости. В свою очередь, даже наиболее изощренные современные численные методы опираются, как правило, на ряд классических математических результатов и представлений, выраженных в базовых методах.

Прямые методы решения систем линейных алгебраических уравнений так или иначе опираются на идеи и приемы понижения

размерности системы путем исключения неизвестных, а также на теоремы о существовании мультипликативных разложений матриц. Чаще всего применяют метод Гаусса, количество операций которого может быть весьма существенно уменьшено при использовании информации о структуре матрицы.

Принцип сжимающих отображений — один из фундаментальных результатов анализа — лежит в основе всего семейства итерационных методов, применяемых для решения систем линейных и нелинейных уравнений, интегральных уравнений и множества других задач. Доказательство сходимости того или иного итерационного метода часто заключается в выяснении условий, при которых оператор перехода между последовательными приближениями является сжимающим.

Наконец, принцип и теоретические обоснования возможности реконструкции функции по дискретному набору ее значений с помощью элементов некоторого функционального пространства позволяют получить семейство методов интерполяции функций, приближенного вычисления определенных интегралов без явного нахождения первообразных, численного решения задач Коши для обыкновенных дифференциальных уравнений.

Перечисленные принципы и результаты позволяют проследить единство изложенных далее методов вычислений.

1. ПРЕДВАРИТЕЛЬНЫЕ СВЕДЕНИЯ

Представлены материалы об источниках погрешностей при вычислениях, об особенностях машинной арифметики и ее результатах. Вычислены погрешности арифметических операций. Даны примеры устойчивых и неустойчивых алгоритмов. Приведены основные сведения из линейной алгебры, теории линейных операторов в конечномерных пространствах и функционального анализа.

1.1. Погрешности при вычислениях

1.1.1. Причины появления погрешностей

При численном решении задач приходится иметь дело не только с теоретическими аспектами существования решения, скорости сходимости и точности выбранного численного метода, но и не в последнюю очередь с практическими вопросами, связанными с реализацией метода на ЭВМ, с заданием исходных данных, подчас известных только приближенно, с особенностями выбранной математической модели.

Задача любого расчета — инженерно-практического или научно-теоретического — заключается в определении некоторой искомой величины y по заданной x . Обычно предполагается существование связи между ними, которая в общем виде может быть записана как $y = A(x)$.

Задача кажется простой и ясной, но в действительности точно вычислить y за исключением тривиальных случаев невозможно. Причин этому несколько.

1. Входные данные x , представляющие собой те или иные физические величины (скорость, давление, температура и т. п.),

измеряются приближенно и никогда (за исключением тривиальной целочисленной информации) точно не известны. Известно лишь \tilde{x} — некоторое приближение x .

2. Ввиду неполноты наших знаний об окружающем мире вместо сложной истинной зависимости A известна лишь некоторая приближенная зависимость \tilde{A} . Поэтому и вместо искомой величины в конечном счете вычисляется некоторая $\tilde{y} = \tilde{A}(\tilde{x})$.

3. При численном решении зависимость \tilde{A} (оператор или функция) заменяется на ее приближение \tilde{A}_h , так что в результате расчета может быть найдена лишь величина $\tilde{y}_h = \tilde{A}_h(\tilde{x})$.

4. Проведение вычислений на ЭВМ в силу особенностей хранения чисел вносит дополнительную погрешность. В результате получается величина $\tilde{y}_h^* = \tilde{A}_h^*(\tilde{x})$.

Таким образом возникают погрешности нескольких видов:

- ***неустранимая погрешность***

$$\rho_1 = \tilde{y} - y = \tilde{A}(\tilde{x}) - A(x)$$

(в рамках данного подхода ρ_1 вызвана неточностью модели и входных данных);

- ***погрешность численного метода***

$$\rho_2 = \tilde{y}_h - \tilde{y} = \tilde{A}_h(\tilde{x}) - \tilde{A}(\tilde{x});$$

- ***погрешность вычислений***

$$\rho_3 = \tilde{y}_h^* - \tilde{y}_h = \tilde{A}_h^*(\tilde{x}) - \tilde{A}_h(\tilde{x}).$$

Общая погрешность найденного результата

$$\rho = \tilde{y}_h^* - y = \tilde{y}_h^* - \tilde{y}_h + \tilde{y}_h - \tilde{y} + \tilde{y} - y = \rho_3 + \rho_2 + \rho_1.$$

Одна из задач разработки численного метода — обеспечить такую точность, чтобы в процессе вычислений величина $\rho_2 + \rho_3$ оказалась в несколько раз меньше ρ_1 .

Отметим, что все перечисленные погрешности носят объективный по отношению к вычислителю характер.

1.1.2. Хранение чисел на ЭВМ и погрешности округления

Рассмотрим погрешности, приводящие к возникновению погрешности вычислений ρ_3 . В любой ЭВМ для хранения чисел используется конечное число разрядов. И если целые числа (с определенными ограничениями) можно хранить в памяти ЭВМ точно, то остальные действительные числа могут быть представлены лишь приближенно. Аналогично и арифметические действия с этими числами выполняются приближенно. Чтобы подчеркнуть этот факт, говорят о вычислениях в **машинной**, или **конечной, арифметике**.

Наиболее распространенная форма представления действительных чисел в ЭВМ — их запись с плавающей запятой.

Определение. **Число с плавающей запятой** — это число вида

$$a = Mr^p,$$

где $|M| < 1$ — число с фиксированной запятой, называемое мантиссой; r — основание системы счисления; p — целое число — порядок числа a .

В качестве основания обычно принимают $r = 2$, поскольку такой выбор дает минимальные погрешности округления. Далее при необходимости (если используется $r \neq 10$ или для подчеркивания различия оснований) будем указывать основание системы счисления нижним индексом при числе, как, например, для числа 110_2 в двоичной системе, равного 6_{10} в десятеричной.

В памяти ЭВМ можно сохранять лишь ограниченное число разрядов мантиссы и порядка. Поэтому в ЭВМ невозможно точно хранить не только трансцендентные числа (бесконечные непериодические десятичные дроби), но и рациональные (например, $1/3$), а также слишком большие или малые по абсолютному

значению. В результате в ЭВМ вместо точного действительного числа a хранится его приближение a^* в виде числа с плавающей запятой.

Определение. *Число с плавающей запятой* называется **нормализованным**, если старший разряд его мантиссы отличен от нуля. В этом случае выполняется неравенство $r^{-1} \leq |M| < 1$.

Например, число $0,10101_2 \cdot 2^3$ нормализовано, а число $0,010101_2 \cdot 2^4$ нет. Обычно числа с плавающей запятой нормализуют, что дает выигрыш в двух отношениях: любое ненулевое число с плавающей запятой в этом случае имеет единственное представление в виде строки битов, и старший разряд двоичной мантиссы можно не хранить в явном виде (поскольку он всегда равен единице); за счет сэкономленного бита можно удлинить мантиссу.

Наиболее важные параметры, описывающие числа с плавающей запятой, — это основание r , число разрядов (битов) мантиссы M , определяющее точность представления, и число разрядов (битов) порядка p , определяющее область изменения показателей и тем самым наибольшее и наименьшее из представимых чисел.

Пусть под порядок отведено m разрядов, а под мантиссу — l разрядов. Тогда числа $\pm r^{r^m}$ определяют левую и правую границу допустимого числового диапазона. При этом число r^{-r^m} представляет собой **машинный нуль**, а r^{r^m-1} — **машинную бесконечность**. Термин «машинный нуль» вводится потому, что числа, меньшие машинного нуля, ЭВМ полагает равными нулю. С числами за пределами машинной бесконечности ЭВМ без специальных процедур работать уже не может.

На практике используют приемы, позволяющие увеличить число хранимых знаков мантиссы и порядка и тем самым расширить пределы представимости чисел в машинной арифметике.

Различные арифметики с плавающей запятой разнятся между собой также способом округления вычисленных результатов, наличием или отсутствием символов $\pm\infty$ и некоторых полезных «нечисел». «Нечисло», называемое еще неопределенной величиной или специальным операндом, иногда обозначается Nan .

Для единообразного представления в памяти ЭВМ чисел с плавающей запятой разработана серия международных стандартов, из которых наиболее часто употребляется IEEE-стандарт двоичной арифметики (табл. 1.1). В IEEE-арифметике предусмотрено два типа чисел с плавающей запятой: одинарной точности (с 32-битовым представлением) и двойной точности (64 бита). В соответствии с этим стандартом для хранения числа с плавающей запятой используются три битовые последовательности (числа) s , e и f .

Таблица 1.1

Представление чисел с плавающей запятой в памяти ЭВМ в соответствии со стандартом IEEE 754

Точность числа	Число битов		
	Знак	Порядок m	Мантисса l
Одинарная	1	8	23
Двойная	1	11	52

В случае одинарной точности число с плавающей запятой представляется в виде

$$a^* = (-1)^s \cdot 2^{e-127} (1+f),$$

где s — однобитовый знак числа; e — 8-битовый порядок; f — 23-битовая мантисса.

Такое представление числа позволяет не хранить в явном виде знак порядка, поскольку максимальное значение порядка $e = 1111111_2 = 255 = 2^8 - 1$, а $127 = 2^7 - 1$ есть примерно половина этого значения. В этом случае порядок может принимать

значения от -127 до 128 . При одинарной точности максимальная относительная погрешность $|a - a^*|/|a^*|$ представления числа a в конечной арифметике в виде a^* равна $2^{-24} \approx 6 \cdot 10^{-8}$, поскольку значение 24-го двоичного знака мантиссы числа a на представлении a^* не отражается. Область положительных нормализованных чисел простирается приблизительно от 10^{-38} до 10^{38} .

В случае двойной точности

$$a^* = (-1)^s \cdot 2^{e-1023} (1+f),$$

при этом порядок e занимает 11 битов, а мантисса f — 52 бита. Аналогичные приведенным выше рассуждения с учетом того, что $1023 = 2^{10} - 1$, а $1111111111_2 = 2^{11} - 1$, дают диапазон изменения порядка от -1023 до 1024 . Максимальная относительная погрешность представления равна $2^{-53} \approx 10^{-16}$, а границами области представимых положительных нормализованных чисел являются приблизительно 10^{-308} и 10^{308} .

Расположение чисел с плавающей запятой на вещественной числовой прямой неравномерно, такие числа чаще встречаются в окрестности нуля и все реже при удалении от него: увеличение порядка на единицу удваивает расстояние между соседними числами (рис. 1.1).

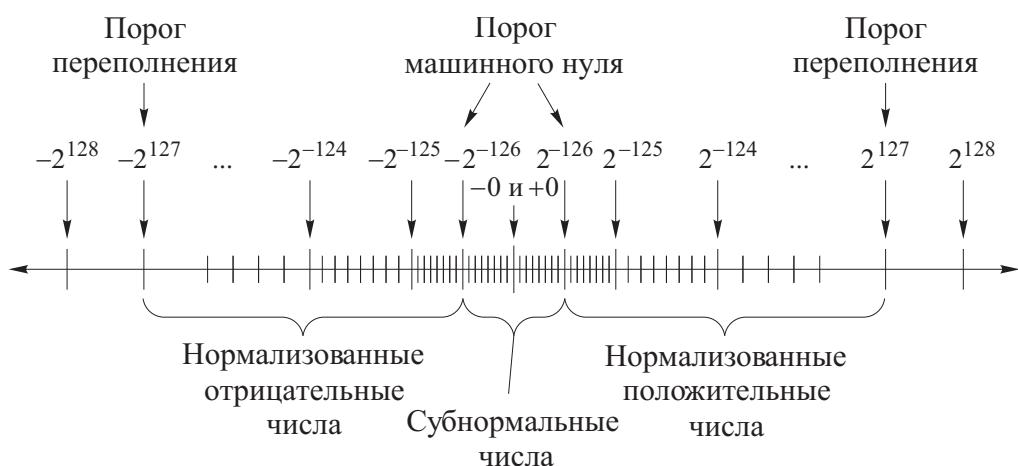


Рис. 1.1. Неравномерность расположения чисел с плавающей запятой на числовой прямой

Отметим, что в настоящее время можно использовать так называемую точную арифметику. Модули для ее реализации уже разработаны для многих прикладных языков программирования (C, C++, Фортран и др.). Она лежит в основе большинства крупных математических программных пакетов. В точной арифметике иррациональные числа представляются как набор предикатов (арифметических и алгебраических операторов) от рациональных или целых чисел и установленных констант (таких как e , π и им аналогичных). Например, в точной арифметике $\sqrt{3}$ — это действительно $\sqrt{3}$, а не $1,73205080756887729\dots$. Недостаток такого подхода — весьма значительное снижение скорости массовых вычислений, так как ни точная арифметика, ни числа с фиксированной запятой пока аппаратно не поддерживаются процессорами современных персональных компьютеров. Поэтому все операции приходится реализовывать на уровне подпрограмм, что приводит к дополнительным затратам ресурсов.

При появлении в процессе счета мантиссы с числом знаков, большим допускаемого системой, происходит округление. Оно может выполняться либо простым отбрасыванием первого лишнего разряда (правило усечения), либо в соответствии со специальными правилами (правило дополнения), либо случайным образом.

Вычисления в конечной арифметике существенно отличаются от точных вычислений, причем это не экстраординарные различия. Подобные погрешности сопровождают все содержательные вычисления и в силу их массовости имеют свойство накапливаться.

Пример 1.1. Рассмотрим вычисление в конечной арифметике с 8-битовой мантиссой разности $c = b - a$ чисел $a = 6,2_{10}$ и $b = 6,3_{10}$. Нижний индекс указывает на основание системы счисления, в которой число записано. В точной арифметике $c = 0,1_{10}$.

В двоичной машинной арифметике представлениями чисел a , b и c являются $a^* = 0,11000110_2 \cdot 2^3$, $b^* = 0,11001001_2 \cdot 2^3$ и $c^* = 0,11001101_2 \cdot 2^{-3}$, но разность $b^* - a^*$ имеет нормализованное представление $\tilde{c}^* = 0,11000000_2 \cdot 2^{-3}$ и отличается от c^* уже в пятом знаке после запятой. Отсутствие учета этой особенности приводит, например, к ошибкам при сравнении числа с плавающей запятой с нулем. •

Таким образом, еще раз убедились, что в вычислениях на ЭВМ почти всегда вместо точного числа a имеем дело лишь с его приближением a^* , подчас весьма неточным.

Определение. Величина $\Delta(a^*)$ такая, что $|a - a^*| \leq \Delta(a^*)$, называется **абсолютной погрешностью (ошибкой)** приближенного значения a^* .

Таким образом, точной записью числа является $a = a^* + \Delta^*(a^*)$, где $\Delta^*(a^*)$ — истинная погрешность приближенного значения a^* . Здесь $|\Delta^*(a^*)| \leq \Delta(a^*)$. Очевидно, что значение $\Delta(a^*)$ может быть выбрано единственным способом, поэтому оно должно быть выбрано наименьшим из всех возможных.

Определение. Величина $\delta(a^*)$ такая, что $\frac{|a - a^*|}{|a^*|} \leq \delta(a^*)$, называется **относительной погрешностью (ошибкой)** приближенного значения a^* :

$$\delta(a^*) = \Delta(a^*)/|a^*|.$$

Зачастую используют другую форму записи:

$$a = a^* \pm \Delta(a^*) = a^* (1 \pm \delta(a^*)),$$

показывая, что a^* есть приближенное значение a .

Как и в случае абсолютной погрешности, для $\delta(a^*)$ должно быть выбрано наименьшее из всех возможных значений, удовлетворяющих неравенству, приведенному в определении относительной погрешности.

Далее аналогично истинной абсолютной погрешности $\Delta^*(a^*)$ будем использовать истинную относительную погрешность

$$\delta^*(a^*) = \frac{\Delta^*(a^*)}{|a^*|}.$$

Ясно, что точность числа a^* характеризуется главным образом относительной погрешностью $\delta(a^*)$.

Пусть приближенное число a^* задано в виде конечной десятичной дроби.

Определение. *Значащими цифрами числа a^** называются все цифры в его записи (или представлении), начиная с первой ненулевой слева.

Определение. *Значащую цифру числа a^** называют *верной*, если абсолютная погрешность числа не превышает половины единицы разряда, соответствующего этой цифре.

Очевидно, что данное определение ориентировано на использование наиболее распространенного способа округления — правила дополнения. Обычно в этом случае сохраняемые цифры оставляют неизменными, если первая слева из отбрасываемых цифр меньше 5. Если же она равна или больше 5, то в младший сохраняемый разряд добавляется единица.

При округлении по правилу дополнения модуль истинной погрешности округления не превышает половины единицы разряда, соответствующего последней оставляемой цифре, а в случае округления по правилу усечения — единицы того же разряда.

В силу распространенности округления по правилу дополнения запись вида $a = a^*$ чаще всего означает наличие абсолютной погрешности $\Delta(a^*)$, равной половине единицы разряда последней цифры в записи числа a^* .

Нетрудно видеть, что в случае, если число a^* содержит N верных значащих цифр, выполнено приближенное равенство $\delta(a^*) \approx 10^{-N}$.

Погрешности округления (как и большинство погрешностей) имеют накопительный эффект, что приводит к увеличению общих погрешностей вычислений.

1.1.3. Погрешности арифметических операций

Получим формулы для погрешностей результатов четырех основных арифметических операций с двумя приближенно заданными числами a и b .

1. Сложение. Имеем следующую цепочку соотношений:

$$a + b = a^* + \Delta^*(a^*) + b^* + \Delta^*(b^*) = a^* + b^* + \Delta^*(a^* + b^*).$$

Отсюда

$$\begin{aligned} \Delta^*(a^* + b^*) &= \Delta^*(a^*) + \Delta^*(b^*); \\ \delta^*(a^* + b^*) &= \frac{|a^*|}{|a^* + b^*|} \delta^*(a^*) + \frac{|b^*|}{|a^* + b^*|} \delta^*(b^*). \end{aligned}$$

В результате получим

$$\begin{aligned} \Delta(a^* + b^*) &= \Delta(a^*) + \Delta(b^*); \\ \delta(a^* + b^*) &= \frac{|a^*|}{|a^* + b^*|} \delta(a^*) + \frac{|b^*|}{|a^* + b^*|} \delta(b^*). \end{aligned}$$

Если a^*, b^* имеют один знак, то

$$\min(\delta(a^*), \delta(b^*)) \leq \delta(a^* + b^*) \leq \max(\delta(a^*), \delta(b^*)).$$

2. Вычитание. Аналогично сложению получим

$$\begin{aligned} \Delta(a^* - b^*) &= \Delta(a^*) + \Delta(b^*); \\ \delta(a^* - b^*) &= \frac{|a^*|}{|a^* - b^*|} \delta(a^*) + \frac{|b^*|}{|a^* - b^*|} \delta(b^*). \end{aligned}$$

В случае если a^* и b^* имеют близкие значения, относительная погрешность результата может оказаться большой ввиду малости знаменателя в последнем соотношении.

3. Умножение. Выполним умножение, опустив в результате слагаемые, содержащие произведения погрешностей (имеющие, как принято говорить, второй порядок малости):

$$ab = (a^* + \Delta^*(a^*)) (b^* + \Delta^*(b^*)) \approx a^*b^* + a^*\Delta^*(b^*) + b^*\Delta^*(a^*).$$

Отсюда

$$\Delta(a^*b^*) = |a^*|\Delta(b^*) + |b^*|\Delta(a^*);$$

$$\delta(a^*b^*) = \delta(a^*) + \delta(b^*).$$

4. Деление. При вычислении частного также ограничимся главными (с точки зрения малости) слагаемыми:

$$\begin{aligned} \frac{a}{b} &= \frac{a^* + \Delta^*(a^*)}{b^* + \Delta^*(b^*)} = \frac{(a^*/b^*) + \Delta^*(a^*)/b^*}{1 + \Delta^*(b^*)/b^*} \approx \\ &\approx \frac{a^*}{b^*} + \frac{\Delta^*(a^*)}{b^*} - \frac{a^*\Delta^*(b^*)}{(b^*)^2}. \end{aligned}$$

Отсюда

$$\Delta\left(\frac{a^*}{b^*}\right) = \frac{\Delta(a^*)}{|b^*|} + \frac{|a^*|\Delta(b^*)}{(b^*)^2};$$

$$\delta\left(\frac{a^*}{b^*}\right) = \delta(a^*) + \delta(b^*).$$

В этих выражениях не учтены возникающие при вычислениях погрешности округления. Наиболее «опасные» операции с точки зрения значения возникающих погрешностей — вычитание близких чисел и деление на малое число. В обоих случаях даже при достаточно точно известных операндах погрешность может быть большой, в первом случае относительная, во втором — абсолютная.

Если не пренебречь слагаемыми второго порядка малости, то в случае умножения

$$\delta(a^*b^*) = \delta(a^*) + \delta(b^*) + \delta(a^*)\delta(b^*).$$

Если поступить аналогично и с делением, то в результате имеем

$$\delta\left(\frac{a^*}{b^*}\right) = \frac{\delta(a^*) + \delta(b^*)}{1 - \delta(b^*)}.$$

В дальнейшем погрешности округлений и операций в явном виде будем учитывать редко. Однако необходимо постоянно помнить о них и учитывать механизмы их появления.

Пример 1.2. Пусть имеются три приближенно заданных числа a, b, c , для которых известны a^*, b^*, c^* и погрешности. Требуется вычислить величины $u = (a - b)/c$ и $v = (a/c) - (b/c)$ и выяснить, в каком случае погрешность будет больше.

Очевидно, что в рамках точной арифметики $u = v$. В случае приближенно заданных чисел это не так.

Воспользовавшись приближенными формулами для относительной погрешности арифметических операций, получим

$$\begin{aligned}\delta(u^*) &= \delta(a^* - b^*) + \delta(c^*) = \\ &= \frac{|a^*|}{|a^* - b^*|} \delta(a^*) + \frac{|b^*|}{|a^* - b^*|} \delta(b^*) + \delta(c^*); \\ \delta(v^*) &= \left[\left| \frac{a^*}{c^*} \right| (\delta(a^*) + \delta(c^*)) + \left| \frac{b^*}{c^*} \right| (\delta(b^*) + \delta(c^*)) \right] \left| \frac{a^*}{c^*} - \frac{b^*}{c^*} \right|^{-1} = \\ &= \frac{|a^*|}{|a^* - b^*|} \delta(a^*) + \frac{|b^*|}{|a^* - b^*|} \delta(b^*) + \frac{|a^*| + |b^*|}{|a^* - b^*|} \delta(c^*),\end{aligned}$$

где

$$u^* = \frac{a^* - b^*}{c^*};$$

$$v^* = \frac{a^*}{c^*} - \frac{b^*}{c^*}.$$

В результате оказывается, что относительная погрешность v^* больше, чем относительная погрешность u^* . Итог в данном случае вполне ожидаемый, так как для вычисления u^* требуется выполнить два действия, а для вычисления v^* — три. Каждое лишнее действие вносит дополнительную погрешность. Однако это только наводящее соображение. Можно привести примеры разных погрешностей и в случае равного количества выполняемых операций. •

1.1.4. Погрешность алгоритма

Определение. Вычислительный **алгоритм** называется **устойчивым**, если в процессе его работы погрешности округления возрастают незначительно, и **неустойчивым** в противном случае.

С учетом введенного ранее обозначения $\tilde{y}_h^* = \tilde{A}_h^*(\tilde{x})$ это определение может быть сформулировано более конкретно: **алгоритм** \tilde{A}_h^* **устойчив**, если существует константа M такая, что $\Delta(\tilde{y}_h^*) \leq M\Delta(\tilde{x})$, причем константа M не зависит от погрешности входных данных $\Delta(\tilde{x})$.

Однако в силу конечности хранимых величин эта константа может быть различной даже при разном способе организации одних и тех же вычислений, не говоря уже о разных способах решения одной и той же задачи.

Пример 1.3. Рассмотрим уравнение

$$x^2 + 2px + q = 0, \quad |q| \ll p^2.$$

Запишем его решение:

$$x_{1,2} = -p \pm \sqrt{p^2 - q}.$$

Пусть $p, q > 0$. Тогда при вычислении величины

$$x_1 = \sqrt{p^2 - q} - p$$

выполняется крайне неточная операция — вычитание близких чисел. Однако величину x_1 можно записать иначе:

$$x_1 = -\frac{q}{\sqrt{p^2 - q} + p}.$$

При использовании такого выражения вычисления оказываются устойчивыми. Впрочем, этот эффект достигается за счет дополнительной операции — деления. •

Пример 1.4. Найдем частичную сумму гармонического ряда

$$S = \sum_{i=1}^{10^7} \frac{1}{i}.$$

При $n \rightarrow \infty$

$$S_n = \sum_{i=1}^n \frac{1}{i} \rightarrow +\infty.$$

Рассмотрим два варианта вычисления суммы.

1. Вычисление по алгоритму

$$S_1 = 1, \quad S_i = S_{i-1} + \frac{1}{i}, \quad i = \overline{2, 10^7};$$

$$S = S_{10^7}.$$

Поскольку $S_n \sim \ln n$ и $\ln 10 \approx 2,3$, то $S_{999999} \approx \ln 10^6 \approx 13,8$. Если $i = 10^6$, то $S_{10^6} \approx 13,8 + 10^{-6} = 13,8$ при использовании типа данных, обеспечивающего семь значащих цифр. При сложении мантисса меньшего числа сдвигается вправо и добавляется к мантиссе большего числа, следовательно, начиная с этого момента накапливаемая сумма не изменяется. Таким образом, в результате работы алгоритма будет получено неточное значение суммы ряда 13,8 вместо истинного значения $S_n \approx 16,6953$.

2. Вычисление по алгоритму

$$S_{10^7+1}^* = 0, \quad S_{i-1}^* = S_i^* + \frac{1}{i-1}, \quad i = \overline{10^7+1, 2};$$

$$S = S_1^*.$$

Такой порядок расчета не содержит операций с резко различающимися числами. Следовательно, в этом алгоритме потери знаков меньшего числа не происходит. •

Пример 1.5. Вычислим последовательность $\{I_n\}$, где

$$I_0 = e^{-1} \int_0^1 e^x dx = 1 - e^{-1};$$

$$I_n = e^{-1} \int_0^1 x^n e^x dx = 1 - nI_{n-1}, \quad n = 1, 2, \dots .$$

1. Вычислим $\{I_n\}$ в соответствии с представленным выражением. В результате получим, что $\Delta(I_n^*) = n! \Delta(I_0^*)$. Здесь звездочки указывают на приближенный характер используемых величин. Поэтому применение прямой рекуррентной формулы даст катастрофическую потерю точности.

2. Рассмотрим обратное рекуррентное соотношение

$$I_{n-1} = \frac{1 - I_n}{n}.$$

Очевидно, что даже если в качестве «начального» значения последовательности взять заведомо неверное $I_N = 0$ для достаточно большого N и «спуститься» по индексу от N до заданного, то полученная погрешность будет намного меньше, чем в первом алгоритме. ●

Приведенные примеры показывают, что часто для вычисления некоторой величины существуют неустойчивые алгоритмы и при наличии устойчивого. Обратное, к сожалению, возможно не всегда. Таким образом, существуют устойчивые и неустойчивые алгоритмы для решения задач, устойчивость решения которых не вызывает особых сомнений. В то же время существуют так называемые некорректно поставленные задачи, для решения которых весьма сложно создать устойчивый алгоритм.

Рассмотрим отдельно **погрешность вычисления функции**. Пусть $y = f(x)$, f — дифференцируемая функция, $dy = f' dx$. При вычислении величины $y = f(x^* + \Delta^*(x^*))$ приближенное значение

функции $y^* = f(x^*)$. Тогда по формуле конечных приращений Лагранжа

$$\Delta(y^*) = |f'_x(x^* + \vartheta\Delta(x^*))| \Delta(x^*),$$

где $-1 < \vartheta < 1$.

Определение. Величина

$$\Delta(y^*) = |f'_x(x^*)| \Delta(x^*)$$

называется **линейной оценкой абсолютной погрешности вычисления функции**.

Определение. Величина

$$\Delta(y^*) = \sup_{-1 < \vartheta < 1} |f'_x(x^* + \vartheta\Delta(x^*))| \Delta(x^*)$$

называется **пределной оценкой абсолютной погрешности вычисления функции**.

На практике чаще всего пользуются линейной оценкой погрешности. Тогда легко решается обратная задача: найти оценку погрешности аргумента, исходя из заданной погрешности функции.

Например,

$$\begin{aligned} y &= \ln x, \quad \Delta(y^*) = \delta(x^*); \\ y &= \operatorname{tg} x, \quad \Delta(y^*) = (1 + \operatorname{tg}^2 x^*) \Delta(x^*) > \Delta(x^*). \end{aligned}$$

Аналогично по формуле конечных приращений Лагранжа оцениваются погрешности вычисления функций многих переменных. Однако обратная задача в этом случае может быть решена лишь при дополнительных предположениях, например при равном вкладе погрешностей каждой переменной в результирующую погрешность функции, либо при равенстве погрешностей каждого аргумента, либо при каком-то ином условии. В противном случае найти несколько неизвестных из одного уравнения единственным образом не представляется возможным.

1.2. Элементы функционального анализа и линейной алгебры

1.2.1. Линейные пространства

Многие задачи, которые будут рассмотрены далее, сводятся к численному решению уравнения

$$Ax = y,$$

где A — известный оператор; x — неизвестная, подлежащая определению; y — входная информация.

Решение x ищется среди элементов некоторого пространства H .

Определение. Множество H называется **линейным пространством** над полем K действительных или комплексных чисел, если для любых его элементов x и y определены операции сложения и умножения на произвольное число $\lambda \in K$. Результаты сложения $z = x + y \in H$ и умножения на число $w = \lambda x \in H$ принадлежат тому же множеству H , а операции удовлетворяют следующим аксиомам ($\forall x, y, z \in H, \forall \lambda, \mu \in K$):

- 1) $x + y = y + x; x + (y + z) = (x + y) + z$ (коммутативность и ассоциативность сложения);
- 2) $\lambda(\mu x) = (\lambda\mu)x$ (ассоциативность умножения на число);
- 3) $\lambda(x + y) = \lambda x + \lambda y; (\lambda + \mu)x = \lambda x + \mu x$ (дистрибутивность умножения относительно сложения);
- 4) в H существует единственный элемент $0: x + 0 = x, \forall x \in H$;
- 5) для любого $x \in H$ существует единственный элемент $-x \in H: x + (-x) = 0$;
- 6) для каждого $x \in H$ выполнено $1 \cdot x = x$.

Элементы линейного пространства также принято называть векторами; элемент 0 называется нулевым вектором или нулем, а элемент $-x$ — вектором, противоположным вектору x .

Определение. **Элементы** $\{x_i\}$, $i = \overline{1, n}$, $x_i \in H$, называются **линейно независимыми**, если из равенства $\sum_{i=1}^n \lambda_i x_i = 0$ следует, что $\lambda_i = 0$, $i = \overline{1, n}$. Если же равенство нулю последней суммы возможно при хотя бы одном $\lambda_i \neq 0$, то исходные **элементы** называются **линейно зависимыми**.

Очевидно, что любая совокупность элементов, включающая нуль, линейно зависимая.

Определение. **Линейное пространство** H называется **n -мерным**, если в H существует n линейно независимых элементов, а любые $n + 1$ элементов линейно зависимы.

Определение. Непустое замкнутое множество H_1 элементов линейного пространства H называется **линейным многообразием**, если вместе с любыми элементами $x_1, x_2, \dots, x_n \in H_1$ множество H_1 содержит и любую их линейную комбинацию $\sum_{i=1}^n \lambda_i x_i$. Замкнутое линейное многообразие называется **подпространством**.

Определение. **Линейное пространство** H называется **нормированным**, если для каждого $x \in H$ определено вещественное число $\|x\|$, называемое **нормой**, которое удовлетворяет следующим условиям:

- 1) $\|x\| \geq 0$, причем $\|x\| = 0$ тогда и только тогда, когда $x = 0$;
- 2) $\forall x, y \in H$ выполнено $\|x + y\| \leq \|x\| + \|y\|$ (неравенство треугольника);
- 3) $\forall x \in H, \forall \lambda \in K$ выполнено $\|\lambda x\| = |\lambda| \|x\|$ (однородность нормы).

Определение. **Последовательность** элементов $\{x_n\} \subset H$ называется **сходящейся** к $x \in H$ (обозначается $x_n \rightarrow x$), если числовая последовательность $\|x - x_n\| \rightarrow 0$ при $n \rightarrow \infty$. Если

$\|x_m - x_n\| \rightarrow 0$ при независимом стремлении $m, n \rightarrow \infty$, то последовательность $\{x_n\}$ называется **фундаментальной**.

Определение. *Линейное нормированное пространство* H называется **полным** или **банаховым**, если любая фундаментальная последовательность элементов H сходится к некоторому элементу из H .

Любое конечномерное линейное нормированное пространство является полным.

Определение. **Нормы** $\|\cdot\|_1$ и $\|\cdot\|_2$, определенные на линейном пространстве H , называются **эквивалентными**, если существуют такие не зависящие от x постоянные $0 < m \leq M$, что для любого элемента $x \in H$

$$m\|x\|_1 \leq \|x\|_2 \leq M\|x\|_1.$$

В конечномерном пространстве любые две нормы эквивалентны. Постоянные m, M из определения эквивалентности норм чаще всего зависят от размерности пространства.

Из сходимости элементов в одной норме следует их сходимость в любой эквивалентной ей норме.

Определение. Пусть каждой паре $x, y \in H$, где H – линейное нормированное пространство, сопоставлено, вообще говоря, комплексное число (x, y) , удовлетворяющее следующим аксиомам:

- 1) $(x, y) = \overline{(y, x)}$ (симметричность);
- 2) $(x + y, z) = (x, z) + (y, z)$ (дистрибутивность);
- 3) $(\lambda x, y) = \lambda(x, y)$ (однородность);
- 4) $(x, x) \geq 0$, причем $(x, x) = 0$ тогда и только тогда, когда $x = 0$.

Тогда число (x, y) называется **скалярным произведением**. Черта над (y, x) здесь означает комплексное сопряжение.

Если в линейном пространстве H задано скалярное произведение, то это пространство является нормированным с нормой $\|x\| = \sqrt{(x, x)}$ (скалярное произведение (x, x) — действительное число). Такая **норма** называется *естественной* или *порожденной скалярным произведением*.

Определение. *Линейное пространство* H с заданным скалярным произведением и, следовательно, нормой $\|x\| = \sqrt{(x, x)}$ называется **унитарным**. Если пространство H действительное, то оно называется **евклидовым**. Полное относительно естественной нормы унитарное пространство называется **гильбертовым**.

Замечание 1.1. Конечномерное унитарное пространство всегда является гильбертовым. В связи с этим часто понятие гильбертова пространства используется только в отношении бесконечномерных пространств.

Для элементов x и y евклидова пространства справедливо **неравенство Коши — Буняковского** (или Коши — Буняковского — Шварца):

$$|(x, y)| \leq \|x\| \|y\|.$$

Определение. Элементы $x, y \in H$ унитарного пространства H называются **взаимно ортогональными** (обозначаются $x \perp y$), если $(x, y) = 0$. Если x ортогонален любому элементу y подпространства H_1 , то говорят, что x ортогонален H_1 : $x \perp H_1$. Множество $H_2 = \{x \in H : x \perp H_1\}$ называется **ортогональным дополнением** H_1 , $H_2 \perp H_1$.

Определение. *Система* $\{x_i\}$, $i = 1, 2, \dots$; $x_i \in H$, называется **ортонормированной**, если для любых индексов i и j выполнено $(x_i, x_j) = \delta_{ij}$ (δ_{ij} — символ Кронекера: $\delta_{ii} = 1$; $\delta_{ij} = 0$, если $i \neq j$). Если в H не существует элемента, отличного от нуля и ортогонального всем x_i , то такая *система* называется **полной**.

1.2.2. Примеры нормированных линейных пространств

Приведем примеры нормированных пространств, важных для изучения методов вычислений.

1. Множество действительных чисел \mathbb{R} со стандартной операцией сложения, нормой $\|x\| = |x|$ и операцией умножения на целое число является линейным нормированным пространством над полем целых чисел.

2. Векторное пространство \mathbb{R}^n с элементами $x = (x_1, \dots, x_n)^\top$, $y = (y_1, \dots, y_n)^\top$ и покомпонентной операцией сложения

$$x + y = (x_1 + y_1, \dots, x_n + y_n)^\top$$

является линейным. Скалярное произведение

$$(x, y) = x_1 y_1 + \dots + x_n y_n$$

вводит евклидову норму $\|x\| = \sqrt{x_1^2 + \dots + x_n^2}$.

3. Пространство $C[a, b]$ непрерывных на отрезке $[a, b]$ функций $x(t)$ является линейным и полным с нормой

$$\|x\|_C = \max_{t \in [a, b]} |x(t)|.$$

4. Пространство $L_2[a, b]$ функций, интегрируемых с квадратом в смысле Лебега, является евклидовым при стандартном понимании сложения:

$$\forall f, g \in L_2[a, b] \quad h = f + g \in L_2[a, b] : h(t) = f(t) + g(t).$$

Скалярное произведение функций f и g определено формулой $(f, g) = \int_a^b f(t)g(t) dt$ и порождает норму $\|f\| = \left(\int_a^b f^2(t) dt \right)^{1/2}$.

5. Пространство $L_p([a, b])$, $1 < p < +\infty$, состоящее из функций $x(t)$, определенных на отрезке $[a, b]$ и удовлетворяющих условию $\int_a^b |x|^p dt < \infty$ (здесь интеграл понимается в смысле

Лебега), называется пространством измеримых функций и является полным. Норма в этом пространстве

$$\|x\|_{L_p} = \left(\int_a^b |x|^p dt \right)^{1/p}.$$

Отметим, что функции, различающиеся лишь на множестве меры нуль, считаются одинаковыми (эквивалентными).

6. Бесконечномерные пространства c и l_p последовательностей вида $x = \{x_n\}$ являются линейными с нормами

$$\|x\|_c = \sup |x_i|; \quad \|x\|_{l_p} = \lim_{n \rightarrow \infty} \left(\frac{1}{n} \sum_{i=1}^n |x_i|^p \right)^{1/p}.$$

7. Рассмотрим линейное пространство \mathbb{R}^n и введем на нем скалярное произведение и норму следующим образом. Пусть h_1, \dots, h_{n-1} — набор из $n-1$ положительных действительных чисел. Введем обозначения:

$$h_i = \frac{1}{2}(h_{i-1} + h_i), \quad i = \overline{2, n-1};$$

$$h_1 = \frac{h_1}{2}; \quad h_n = \frac{h_{n-1}}{2}.$$

Тогда величина

$$[x, y] = \sum_{i=1}^n x_i y_i h_i$$

есть скалярное произведение векторов x и y , а $|[x]| = \sqrt{[x, x]}$ — норма вектора x . Симметричность, дистрибутивность и однородность скалярного произведения очевидны. Покажем положительную определенность скалярного произведения векторов:

$$[x, x] = \sum_{i=1}^n x_i x_i h_i \geq \min_i h_i \sum_{i=1}^n x_i^2 = (x, x) \min_i h_i \geq 0,$$

причем $[x, x] = 0$ тогда и только тогда, когда $(x, x) = 0$, или $x = 0$.

8. **Пространство Соболева** $W_2^1([a, b])$ представляет собой пополнение* нормированного пространства бесконечно дифференцируемых на отрезке $[a, b]$ функций с нормой

$$\|y\|_{W_2^1} = \left(\int_a^b \left(y^2(t) + (y'(t))^2 \right) dt \right)^{1/2}.$$

Пополнение нормированного пространства финитных бесконечно дифференцируемых функций с этой нормой приводит к полному нормированному пространству $W_2^{1,0}$. В этом нормированном пространстве можно упростить выражение для нормы, опустив под знаком интеграла первое слагаемое.

1.2.3. Операторы в нормированных пространствах

Пусть X, Y — линейные нормированные пространства над полем K . Говорят, что на множестве $D \subset X$ задан оператор A со значениями в пространстве Y , если любому элементу $x \in D$ сопоставлен элемент $y = Ax \in Y$. При этом $D = D(A)$ — **область определения оператора** A , $\text{im } A = \{y : y = Ax, x \in D(A)\}$ — **область значений оператора** A . Если $D(A) = X$, $\text{im } A \subset Y$, то A отображает X на себя, т. е. A — оператор (действующий) на X .

Будем обозначать через E единичный оператор ($x = Ex$), а через 0 — нулевой оператор ($0x = 0$).

Определение. *Оператор* A называется **линейным**, если $D(A)$ — линейное многообразие в пространстве X и

$$A(\lambda x + \mu y) = \lambda Ax + \mu Ay, \quad x, y \in D(A); \quad \lambda, \mu \in K.$$

*Процедуру пополнения нормированного пространства можно рассматривать как расширение этого линейного пространства добавлением формальных пределов фундаментальных последовательностей. Детали этой процедуры в данном издании не приводятся, поскольку выходят за рамки изучаемой дисциплины.

Определение. *Линейный оператор* A называется *ограниченным*, если

$$\exists M > 0 : \forall x \in D(A) \quad \|Ax\|_2 \leq M\|x\|_1,$$

где $\|\cdot\|_1, \|\cdot\|_2$ — нормы в пространствах X и Y соответственно. Наименьшая из постоянных M называется *нормой оператора* A , *подчиненной* норме $\|\cdot\|_1$:

$$\|A\| = \sup_{\substack{x \in D(A) \\ x \neq 0}} \frac{\|Ax\|_2}{\|x\|_1} = \sup_{\substack{x \in D(A) \\ \|x\|_1=1}} \|Ax\|_2.$$

Из определения подчиненной нормы следует оценка

$$\|Ax\|_2 \leq \|A\|\|x\|_1.$$

Определение. *Оператор* A называется *непрерывным* в точке x , если из $\|x_n - x\|_1 \rightarrow 0$ следует $\|Ax_n - Ax\|_2 \rightarrow 0$.

Линейный ограниченный оператор всегда непрерывен и, наоборот, линейный непрерывный оператор ограничен.

Всевозможные линейные ограниченные операторы из X в Y образуют линейное нормированное пространство.

На множестве линейных ограниченных операторов из X в X ($D(A) = D(B) = X$) можно ввести также операцию умножения по правилу

$$AB(x) = A(B(x)),$$

где AB — линейный ограниченный оператор (композиция операторов A и B). Для введенной подчиненной нормы оператора справедливо неравенство

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

Действительно,

$$\|AB\| = \sup_{x \neq 0} \frac{\|ABx\|_1}{\|x\|_1} \leq \sup_{x \neq 0} \frac{\|A\|\|Bx\|_1}{\|x\|_1} \leq \|A\|\|B\|.$$

Индекс «1» здесь соответствует тому факту, что операторы A и B действуют из X в X .

Определение. Если для всех $x \in X$ выполнено равенство $(AB)x = (BA)x$, то **операторы** A и B называются **перестановочными** или **коммутирующими**.

Определение. Если для любого $y \in Y$ существует единственный $x \in X$, для которого $Ax = y$, то таким соотвествием определяется **оператор** A^{-1} , называемый **обратным** для A и имеющий область определения Y и область значений X , при этом

$$A^{-1}Ax = x; \quad AA^{-1}y = y.$$

Определение. **Ядром** линейного **оператора** A называется множество тех элементов $x \in X$, для которых $Ax = 0$. Ядро линейного оператора обозначается $\ker A$.

Необходимое и достаточное условие для того, чтобы A имел обратный оператор — отсутствие нетривиальных $x \in X$, для которых $Ax = 0$, или $\ker A = \{0\}$. Ядро линейного оператора, действующего в X , — линейное подпространство в X . Сумма размерностей области значений и ядра оператора A , действующего в X , равна размерности X .

Определение. Число

$$\rho(A) = \lim_{k \rightarrow \infty} \|A^k\|^{1/k}$$

называется **спектральным радиусом** **оператора** A . При этом $\rho(A)$ не зависит от выбора нормы и

$$\rho(A) = \inf_{\|\cdot\|} \|A\|.$$

Для любого линейного ограниченного оператора справедливо неравенство

$$\rho(A) \leq \|A\|.$$

Если операторы A и B коммутируют (являются перестановочными), то выполняются также неравенства

$$\rho(AB) \leq \rho(A)\rho(B); \quad \rho(A+B) \leq \rho(A) + \rho(B).$$

Пусть оператор A (не обязательно линейный) отображает X на себя, причем $\text{im } A \subset D(A)$.

Определение. *Оператор* $A: X \rightarrow X$ называется *сжимающим*, если существует такое $\alpha \in [0; 1)$, что для любых $x, y \in X$ имеет место неравенство

$$\|Ax - Ay\| \leq \alpha \|x - y\|.$$

Для сжимающих операторов справедлива теорема Банаха о неподвижной точке, называемая также принципом сжимающих отображений. Приведем ее без доказательства.

Теорема 1.1 (принцип сжимающих отображений). Пусть X — полное нормированное линейное пространство, A — сжимающий оператор на нем. Тогда существует, и притом единственная, *неподвижная точка* x^* этого *оператора*, т. е. такой элемент $x^* \in X$, что $Ax^* = x^*$.

1.2.4. Операторы в гильбертовых пространствах

Из неравенства Коши — Буняковского (см. 1.2.1) следует неравенство

$$|(Ax, x)| \leq \|A\| \|x\|^2.$$

Определение. Ограниченный *оператор* A^* называется *сопряженным* оператору A , если

$$(Ax, y) = (x, A^*y), \quad x, y \in H.$$

Если $A = A^*$, то A — *самосопряженный оператор*.

Для любого линейного ограниченного оператора $\|A^*\| = \|A\|$.

Определение. Оператор A называется **нормальным**, если $AA^* = A^*A$, и **кососимметричным**, если $A^* = -A$.

Любой оператор A можно представить в виде суммы самосопряженного A_1 и кососимметричного A_2 операторов:

$$A = A_1 + A_2, \quad A_1 = (A + A^*)/2; \quad A_2 = (A - A^*)/2.$$

Один из них может быть нулевым.

Если H — действительное пространство, то имеют место соотношения

$$(Ax, x) = (A_1x, x); \quad (A_2x, x) = 0.$$

Для нормального (в частности, самосопряженного) оператора $\rho(A) = \|A\|$, так как для таких операторов справедливо равенство $\|A^k\| = \|A\|^k$. Здесь используется подчиненная норма (см. 1.2.3).

Определение. Линейный **оператор** A в действительном гильбертовом пространстве H называется **положительным** ($A > 0$), если

$$\forall x \in H, x \neq 0 \quad (Ax, x) > 0,$$

неотрицательным ($A \geq 0$), если

$$\forall x \in H \quad (Ax, x) \geq 0,$$

положительно определенным, если

$$\exists \delta > 0: \forall x \in H \quad (Ax, x) \geq \delta(x, x).$$

Отметим, что свойство знакоопределенности полностью зависит от самосопряженной части оператора. В случае комплексного линейного пространства H определение положительного оператора распространяется только на самосопряженные операторы.

Понятие положительного (неотрицательного) оператора вводит в пространстве линейных операторов отношение порядка: неравенство $A > B$ ($A \geq B$) означает, что $A - B > 0$ ($A - B \geq 0$).

Пусть D — самосопряженный положительно определенный оператор в H . Тогда можно ввести понятие **энергетического пространства** H_D , состоящего из элементов пространства H со скалярным произведением $(x, y)_D = (Dx, y)$ и порожденной им нормой $\|x\|_D = (Dx, x)^{1/2}$. Если D — самосопряженный ограниченный положительно определенный в H оператор, то обычная норма $\|\cdot\|$ и энергетическая норма $\|\cdot\|_D$ эквивалентны.

Определение. Числа $\delta = \inf_{\|x\|=1} (Ax, x)$, $\Delta = \sup_{\|x\|=1} (Ax, x)$ называются **границами оператора** A , причем

$$\delta E \leq A \leq \Delta E,$$

где E — единичный оператор.

Определение. Оператор B называется **квадратным корнем из оператора** A , если имеет место равенство $B^2 = BB = A$. Тогда пишут $B = A^{1/2}$.

Можно показать, что существует единственный неотрицательный самосопряженный квадратный корень из любого неотрицательного самосопряженного оператора A , перестановочный с любым оператором, перестановочным с A .

Приведем без доказательства следующие теоремы.

Теорема 1.2. Если A — положительно определенный оператор: $A \geq \delta E$, $\delta > 0$, то существует обратный оператор A^{-1} и $\|A^{-1}\| \leq 1/\delta$.

Теорема 1.3. Пусть A и B — самосопряженные положительно определенные в пространстве H операторы. Тогда неравенства

$$\gamma_1 B \leq A \leq \gamma_2 B;$$

$$\gamma_1 A^{-1} \leq B^{-1} \leq \gamma_2 A^{-1},$$

где $\gamma_2 \geq \gamma_1 > 0$, эквивалентны.

1.2.5. Операторы в конечномерных пространствах

Рассмотрим n -мерное унитарное (или евклидово) пространство H с ортонормированным базисом x_1, x_2, \dots, x_n , причем

$$x = \sum_{k=1}^n c_k x_k, \quad c_k = (x, x_k).$$

Пусть в пространстве H действует линейный оператор A . В ортонормированном базисе x_1, x_2, \dots, x_n ему соответствует матрица $\tilde{A} = (a_{ij})_{n \times n}$, где $a_{ij} = (Ax_j, x_i)$. Действительно,

$$Ax = \sum_{k=1}^n c_k Ax_k = \sum_{l=1}^n d_l x_l,$$

откуда

$$d_l = \sum_{k=1}^n c_k (Ax_k, x_l) = \sum_{k=1}^n a_{lk} c_k, \quad a_{lk} = (Ax_k, x_l).$$

Такая матрица, как видно из построения, единственна, поэтому в практических вопросах принято отождествлять оператор с его матрицей и переносить соответствующие понятия из теории операторов на матрицы.

Любому элементу $x \in H$ соответствует вектор его координат:

$$x = (c_1, c_2, \dots, c_n).$$

Если оператор A — самосопряженный в пространстве H , то соответствующая ему матрица \tilde{A} — симметричная (самосопряженная) в любом ортонормированном базисе.

Определение. Число λ называется **собственным значением (числом)** оператора A , если уравнение $Ax = \lambda x$ имеет нетривиальные решения. При этом x — **собственный элемент (вектор)** оператора A , соответствующий собственному значению λ . Если λ — собственное значение, то $\ker(A - \lambda E) \neq \{0\}$. Множество $\sigma(A)$ собственных значений оператора A называется **спектром оператора A** .

Укажем некоторые **свойства операторов**, имеющие отношение к их спектру.

1. Самосопряженный оператор A имеет n ортонормированных собственных элементов x_1, x_2, \dots, x_n , соответствующих вещественным собственным значениям $\lambda_1, \lambda_2, \dots, \lambda_n$ (спектр самосопряженного оператора является вещественным).

2. Для самосопряженного оператора A

$$\|A\| = \rho(A) = \max_{1 \leq k \leq n} |\lambda_k|.$$

3. Если $A = A^* \geq 0$, то все собственные значения оператора A неотрицательны и, кроме того,

$$\delta(x, x) \leq (Ax, x) \leq \Delta(x, x),$$

где $\delta = \min_k \lambda_k$; $\Delta = \max_k \lambda_k$.

Самосопряженная матрица неотрицательна в том и только том случае, когда все ее собственные значения неотрицательны. Она положительна в том и только том случае, когда все ее собственные значения положительны.

Очевидно, что в последнем случае свойства положительности и положительной определенности операторов совпадают, а роль δ , согласно определению знакоопределенности, играет наименьшее из собственных значений.

Отметим, что в силу однозначности представления матрицы в виде суммы ее симметричной (самосопряженной) и кососимметричной частей свойство знакоопределенности полностью определяется самосопряженной частью. Таким образом, эквивалентность свойств положительности и положительной определенности матрицы распространяется и на случай матриц общего вида.

Определение. Число λ называется **собственным значением** оператора A относительно оператора B , если уравнение

$Ax = \lambda Bx$ имеет ненулевые решения. При этом x — **собственный элемент** оператора A относительно оператора B , соответствующий собственному значению λ .

Если операторы A и B самосопряжены в пространстве H и оператор B положительно определен, то существуют n собственных элементов оператора A относительно оператора B , ортонормированных в энергетическом пространстве H_B .

1.2.6. Нормы векторов и матриц

Рассмотрим конечномерное пространство H размерности n и действующий в нем линейный ограниченный оператор A .

Везде в дальнейшем, если не указано особо, будем отождествлять оператор A с его матрицей \tilde{A} , а вектор — с его координатами. Тильду в обозначении матрицы \tilde{A} будем опускать. В силу такого отождествления наибольший интерес представляют нормы матриц, подчиненные векторным нормам, так как устанавливается прямая связь между понятиями подчиненной нормы оператора и нормы матрицы. При изложении теории методов вычислений чаще всего будем предполагать, что норма матрицы является подчиненной.

Найдем выражение для подчиненных норм матрицы A при различном выборе норм вектора x . Будем полагать, что A действует из X в X , способы вычисления норм $\|x\|$ и $\|Ax\|$ совпадают.

1. **Кубическая норма** $\|x\|_\infty = \max_k |x_k|$. Множество $\|x\|_\infty \leq 1$ представляет собой куб со стороной длиной 2 (на рис. 1.2 изображен «шар» радиусом 1 в кубической норме).

Утверждение 1.1. Справедливо соотношение

$$\|A\|_\infty = \max_i \sum_{j=1}^n |a_{ij}|.$$

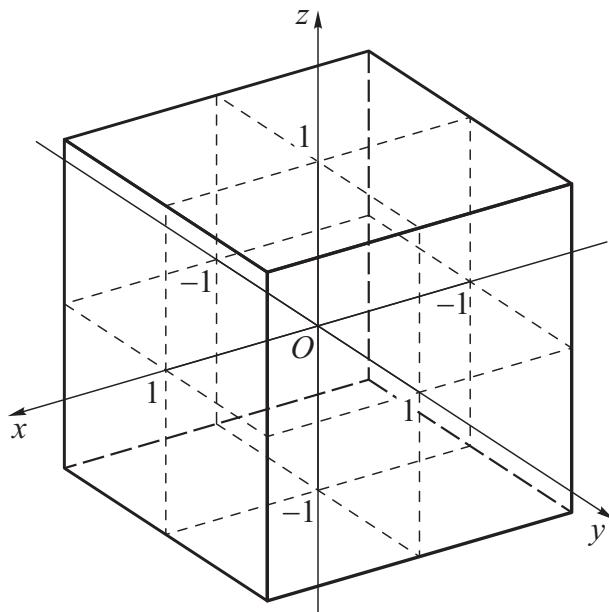


Рис. 1.2. Множество $\|x\|_\infty \leq 1$

◀ По определению

$$\begin{aligned} \|Ax\|_\infty &= \max_i \left| \sum_{j=1}^n a_{ij} x_j \right| \leq \max_i \sum_{j=1}^n |a_{ij}| |x_j| \leq \\ &\leq \max_i \sum_{j=1}^n |a_{ij}| \max_j |x_j| \leq \|x\|_\infty \max_i \sum_{j=1}^n |a_{ij}|. \end{aligned}$$

Найдем вектор $x \neq 0$, для которого это неравенство обращается в равенство. Пусть

$$b_i = \sum_{j=1}^n |a_{ij}|; \quad b_m = \max_i b_i.$$

Выберем $x_j = \text{sign } a_{mj}$, $j = \overline{1, n}$, тогда $\|x\|_\infty = 1$ и $|(Ax)_m| = \sum_{j=1}^n |a_{mj}|$, а для всех остальных компонент

$$|(Ax)_i| = \left| \sum_{j=1}^n a_{ij} x_j \right| \leq \sum_{j=1}^n |a_{ij}| \cdot 1 \leq b_m.$$

Следовательно,

$$\|Ax\|_\infty = \max_i \sum_{j=1}^n |a_{ij}| \cdot 1 = \max_i \sum_{j=1}^n |a_{ij}| = \|A\|_\infty. \quad \blacktriangleright$$

2. Октаэдрическая норма $\|x\|_1 = \sum_{i=1}^n |x_i|$. Множество $\|x\|_1 \leq 1$ представляет собой октаэдр (на рис. 1.3 изображен «шар» радиусом 1 в октаэдрической норме).

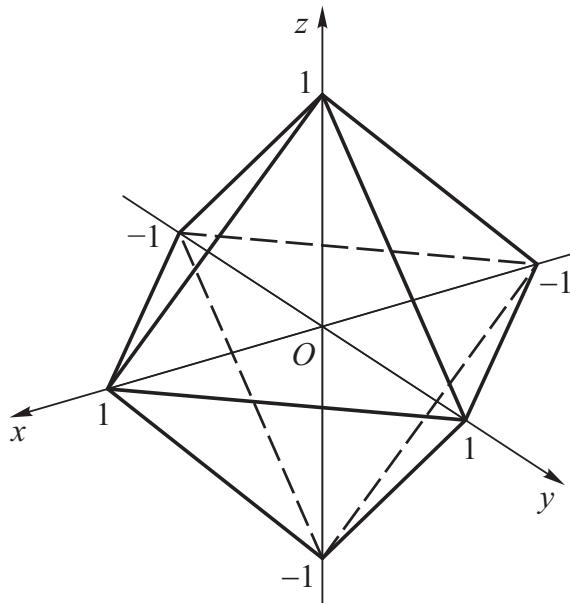


Рис. 1.3. Множество $\|x\|_1 \leq 1$

Утверждение 1.2. Справедливо соотношение

$$\|A\|_1 = \max_j \sum_{i=1}^n |a_{ij}|.$$

◀ По определению

$$\begin{aligned} \|Ax\|_1 &= \sum_{i=1}^n \left| \sum_{j=1}^n a_{ij} x_j \right| \leq \sum_{i=1}^n \sum_{j=1}^n |a_{ij}| |x_j| = \\ &= \sum_{j=1}^n |x_j| \sum_{i=1}^n |a_{ij}| \leq \|x\|_1 \max_j \sum_{i=1}^n |a_{ij}|. \end{aligned}$$

Найдем вектор x , на котором выполнено равенство

$$\|Ax\|_1 = \|x\|_1 \max_j \sum_{i=1}^n |a_{ij}|.$$

Пусть $c_j = \sum_{i=1}^n |a_{ij}|$, $c_p = \max_j c_j$. Выберем $x = \{\delta_{jp}\}$, т. е. единица в строке p и нуль в остальных местах. Тогда $\|x\|_1 = 1$,

$$(Ax)_i = \sum_{j=1}^n a_{ij}x_j = a_{ip}.$$

Следовательно,

$$\|Ax\|_1 = \sum_{i=1}^n |a_{ip}| = c_p = \max_j \sum_{i=1}^n |a_{ij}| = \|A\|_1. \quad \blacktriangleright$$

3. **Евклидова норма** $\|x\|_2 = \left(\sum_{i=1}^n x_i^2 \right)^{1/2}$ (иногда ее называют **гильбертовой** или **шаровой нормой**). Множество $\|x\|_2 \leq 1$ — обычный шар радиусом 1 в декартовых координатах (на рис. 1.4 приведен привычный шар радиусом 1 в евклидовой норме), отсюда и название.

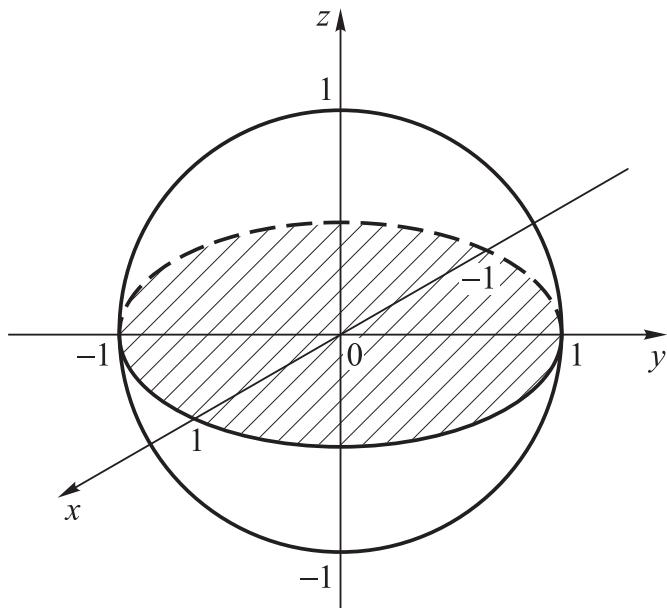


Рис. 1.4. Множество $\|x\|_2 \leq 1$

Утверждение 1.3. Норма матрицы $\|A\|_2 = \left(\sum_{i,j=1}^n a_{ij}^2 \right)^{1/2}$ не является нормой, подчиненной сферической норме вектора.

◀ По определению

$$\|Ax\|_2 = \left(\sum_{i=1}^n \left(\sum_{j=1}^n a_{ij} x_j \right)^2 \right)^{1/2},$$

но вследствие неравенства Коши — Буняковского

$$\left| \sum_{j=1}^n a_{ij} x_j \right| \leq \left(\sum_{j=1}^n a_{ij}^2 \right)^{1/2} \left(\sum_{j=1}^n x_j^2 \right)^{1/2}$$

и тем самым

$$\|Ax\|_2 \leq \left(\sum_{i,j=1}^n a_{ij}^2 \right)^{1/2} \|x\|_2.$$

Далее следует указать вектор x , для которого последнее неравенство обращается в равенство. Однако это невозможно, так как при $n \neq 1$, например, для единичного оператора E норма

$$\|E\|_2 = \left(\sum_{i,j=1}^n \delta_{ij}^2 \right)^{1/2} = \sqrt{n}; \quad \|Ex\|_2 = \|x\|_2 \neq \sqrt{n} \|x\|_2 \quad \forall x \neq 0.$$

Подчиненная норма оператора E

$$\|E\| = \sup_{x \neq 0} \frac{\|Ex\|_2}{\|x\|_2} = 1.$$

Таким образом, $\|A\|_2$ не является нормой матрицы, подчиненной норме вектора $\|\cdot\|_2$. Однако $\|A\|_2$ — это величина, для которой выполнены все три условия нормы (см. 1.2.1), т. е. $\|A\|_2$ — норма в пространстве матрицы. Она называется **нормой Фробениуса матрицы**. ▶

Определение. ***Норма матрицы*** $\|A\|$ называется ***согласованной*** с нормой вектора $\|x\|_X$, если

$$\|Ax\|_Y \leq \|A\| \|x\|_X, \quad x \in H.$$

Все подчиненные нормы матрицы согласованы с соответствующими нормами вектора. Так, нормы матрицы $\|\cdot\|_1$, $\|\cdot\|_\infty$, $\|\cdot\|_2$ — согласованные с одноименными нормами вектора.

Для подчиненной нормы матрицы, определяемой соотношением

$$\|A\| = \sup_{x \neq 0} \frac{\|Ax\|_1}{\|x\|_2},$$

справедливо неравенство $\|AB\| \leq \|A\| \|B\|$. Оно выполнено для произвольных матриц A и B и для нормы Фробениуса $\|A\|_2$.

Однако это может быть не так для других кандидатов на роль нормы, удовлетворяющих условиям нормы в определении нормированного линейного пространства (см. 1.2.1).

Утверждение 1.4. Для величины $\tilde{N}(A) = \max_{i,j} |a_{ij}|$ условия нормы справедливы, а неравенство $\tilde{N}(AB) \leq \tilde{N}(A)\tilde{N}(B)$, вообще говоря, не выполнено.

◀ По построению $\tilde{N}(A) \geq 0$. Если $\tilde{N}(A) = 0$, то $a_{ij} = 0 \quad \forall i, j$ и, следовательно, $A = 0$. При $\alpha \in \mathbb{R}$

$$\tilde{N}(\alpha A) = \max_{i,j} |\alpha a_{ij}| = |\alpha| \max_{i,j} |a_{ij}| = |\alpha| \tilde{N}(A).$$

Очевидно также, что $\tilde{N}(A+B) \leq \tilde{N}(A) + \tilde{N}(B)$. Тем не менее при

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

имеем $\tilde{N}(A) = 1$; $\tilde{N}(A^2) = 2$ и, следовательно, $\tilde{N}(A^2) > (\tilde{N}(A))^2$. ►

В связи с важностью неравенства $\|AB\| \leq \|A\| \|B\|$ его включают в определение нормы матрицы.

Определение. Число $\|A\|$ называется **матричной нормой** матрицы A , если выполняются следующие условия:

- 1) $\|A\| \geq 0$ для всех A ; $\|A\| = 0$ тогда и только тогда, когда $A = 0$;
- 2) $\|A + B\| \leq \|A\| + \|B\|$ для любых матриц A и B (неравенство треугольника);
- 3) $\|\lambda A\| = |\lambda| \|A\|$ для любой матрицы A и любого $\lambda \in K$ (однородность нормы);
- 4) $\|AB\| \leq \|A\| \|B\|$ для любых матриц A и B .

Далее всегда будем использовать только матричные нормы. Таким образом, величина $\tilde{N}(A)$, определенная в утверждении 1.4, не является матричной нормой, хотя удовлетворяет всем аксиомам норм.

Используемые далее нормы будут, как правило, подчиненными или как минимум согласованными.

Утверждение 1.5. Величина $\|A\|_M = n \max_{i,j} |a_{ij}|$ согласована с $\|x\|_1$, $\|x\|_2$, $\|x\|_\infty$ и является нормой.

◀ Покажем только согласованность $\|A\|_M$ с евклидовой нормой вектора:

$$\begin{aligned} \|Ax\|_2 &= \sqrt{\sum_{i=1}^n \left(\sum_{j=1}^n a_{ij} x_j \right)^2} \leq \max_{i,j} |a_{ij}| \sqrt{\sum_{i=1}^n \left(\sum_{j=1}^n |x_j| \right)^2} = \\ &= \sqrt{n} \max_{i,j} |a_{ij}| \left| \sum_{j=1}^n |x_j| \right| = \sqrt{n} \max_{i,j} |a_{ij}| \sum_{j=1}^n |x_j| \leq \\ &\leq \sqrt{n} \max_{i,j} |a_{ij}| \left(\sum_{j=1}^n 1^2 \right)^{1/2} \left(\sum_{j=1}^n x_j^2 \right)^{1/2} = \\ &= n \max_{i,j} |a_{ij}| \|x\|_2 = \|A\|_M \|x\|_2. \end{aligned}$$

Для доказательства использовано неравенство Коши — Буняковского в виде

$$\left| \sum_{j=1}^n 1 \cdot x_j \right| \leq \left(\sum_{j=1}^n 1^2 \right)^{1/2} \left(\sum_{j=1}^n x_j^2 \right)^{1/2}. \quad \blacktriangleright$$

Данную *норму матрицы* часто называют *максимальной*. Отметим, что в отличие от функции $\tilde{N}(A)$ из утверждения 1.4 эта норма зависит от размерности n пространства.

Утверждение 1.6. Величина $\tilde{\tilde{N}}(A) = \frac{1}{\sqrt{n}} \|A\|_2$ не является матричной нормой.

◀ Покажем это, приведя контрпример. Пусть

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

Тогда $\tilde{\tilde{N}}(A) = \frac{1}{\sqrt{2}}$ и

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

Однако $\tilde{\tilde{N}}(AA) = \frac{1}{\sqrt{2}} > \tilde{\tilde{N}}(A) \tilde{\tilde{N}}(A) = \frac{1}{2}$, а значит, четвертое условие из определения матричной нормы не выполнено. ►

Утверждение 1.7. Пусть $\|A\|_s = \left(\max_j \mu_j \right)^{1/2}$, где μ_j — собственные значения матрицы A^*A , $\det(A^*A - \mu E) = 0$. Величина $\|\cdot\|_s$ подчинена векторной норме $\|\cdot\|_2$ и является нормой. Числа μ_j называют *сингулярными числами* матрицы A . Очевидно, всегда $\mu_j \geq 0$.

◀ Покажем подчиненность матричной нормы $\|\cdot\|_s$ векторной норме $\|\cdot\|_2$. Оператор A^*A — самосопряженный и неотрицательный: $(A^*Ax, x) = (Ax, Ax) = \|Ax\|_2^2 \geq 0$. В силу этого он имеет

полную систему собственных векторов и собственных значений, причем любой вектор x можно разложить в сумму:

$$x = \sum_{i=1}^n c_i x_i,$$

где x_i — собственные векторы оператора A^*A . Отметим, что

$$\|x\|_2 = \sqrt{\sum_{i=1}^n c_i^2}; \quad (A^*Ax, x) = \sum_{i=1}^n \mu_i c_i^2,$$

где μ_i — собственные значения оператора A^*A .

По определению подчиненной нормы

$$\begin{aligned} \|A\| &= \sup_{x \neq 0} \frac{\|Ax\|_2}{\|x\|_2} = \sup_{x \neq 0} \sqrt{\frac{(Ax, Ax)}{(x, x)}} = \\ &= \sup_{x \neq 0} \sqrt{\frac{(A^*Ax, x)}{(x, x)}} = \sup_{x \neq 0} \left(\frac{\sum_{i=1}^n \mu_i c_i^2}{\sum_{i=1}^n c_i^2} \right)^{1/2} \leq \|A\|_s. \end{aligned}$$

В последнем нестрогом неравенстве равенство реализуется, если в качестве вектора x под знаком точной верхней грани взять собственный вектор оператора A^*A , соответствующий максимальному собственному значению. Таким образом, матричная норма $\|\cdot\|_s$ подчинена векторной евклидовой норме. ►

Данная **норма матрицы** $\|\cdot\|_s$ называется **спектральной**. Если $A^* = A$, то μ_j равны квадратам собственных значений A , т. е. $\|A\|_s = \max_j |\lambda_j|$, или $\|A\|_s = \rho(A)$, где λ_j — собственные значения A ; $\rho(A)$ — спектральный радиус A .

Пространство матриц размерностью $n \times n$ — конечномерное, поэтому все вышеперечисленные нормы эквивалентны. Покажем это на следующих примерах.

Пример 1.6. Докажем, что

$$\frac{1}{n} \|A\|_M \leq \|A\|_\infty \leq \|A\|_M.$$

Действительно,

$$\frac{1}{n} \|A\|_M = \max_{i,j} |a_{ij}| = \max_i \max_j |a_{ij}| \leq \max_i \sum_{j=1}^n |a_{ij}| = \|A\|_\infty;$$

$$\|A\|_\infty = \max_i \sum_{j=1}^n |a_{ij}| \leq n \max_{i,j} |a_{ij}| = \|A\|_M. \bullet$$

Пример 1.7. Докажем неравенство

$$\frac{1}{\sqrt{n}} \|A\|_2 \leq \|A\|_\infty \leq \sqrt{n} \|A\|_2.$$

Действительно, применяя неравенство Коши — Буняковского, получим

$$\begin{aligned} \|A\|_\infty &= \max_i \sum_{j=1}^n |a_{ij}| \leq \max_i \left(\sum_{j=1}^n 1^2 \sum_{j=1}^n a_{ij}^2 \right)^{1/2} = \\ &= \sqrt{n} \max_i \left(\sum_{j=1}^n a_{ij}^2 \right)^{1/2} \leq \sqrt{n} \left(\sum_{i,j=1}^n a_{ij}^2 \right)^{1/2} = \sqrt{n} \|A\|_2; \end{aligned}$$

$$\begin{aligned} \frac{1}{\sqrt{n}} \|A\|_2 &= \frac{1}{\sqrt{n}} \left(\sum_{i,j=1}^n a_{ij}^2 \right)^{1/2} \leq \frac{1}{\sqrt{n}} \left(\max_i n \sum_{j=1}^n a_{ij}^2 \right)^{1/2} = \\ &= \max_i \left(\sum_{j=1}^n a_{ij}^2 \right)^{1/2} \leq \max_i \sum_{j=1}^n |a_{ij}| = \|A\|_\infty. \bullet \end{aligned}$$

Понятие нормы матриц используется как при доказательстве тех или иных теоретических положений о методах вычислений, так и в практических расчетах, например при оценке числа обусловленности матрицы в задаче о решении системы линейных уравнений. В первом случае эквивалентность норм позволяет применить любую удобную для данного конкретного рассуждения

ния норму. Во втором случае выбор той или иной нормы может оказывать влияние на результат вычислений. Это влияние носит, как правило, не качественный, а количественный характер. Однако чтобы разработать более эффективный вычислительный алгоритм, может быть удобно использовать несколько различных норм для построения оценок одной и той же величины.

1.2.7. Геометрическая интерпретация понятия линейного оператора

Рассмотрим для наглядности случай, когда линейный оператор задан на пространстве $H = \mathbb{R}^3$. Будем интерпретировать это пространство как множество трехмерных вещественных векторов с точкой приложения в начале координат O . Частными случаями линейных операторов, действующих в таком пространстве, являются операторы поворота, растяжения (подобия), отражения относительно прямой или плоскости, проходящих через точку O , и др.

Действие самосопряженного положительно определенного оператора A на произвольный вектор пространства H представляет собой комбинацию поворота и различного для разных векторов растяжения, причем тройка базисных ортов переходит в тройку, вообще говоря, не единичных и не ортогональных линейно независимых векторов (рис. 1.5).

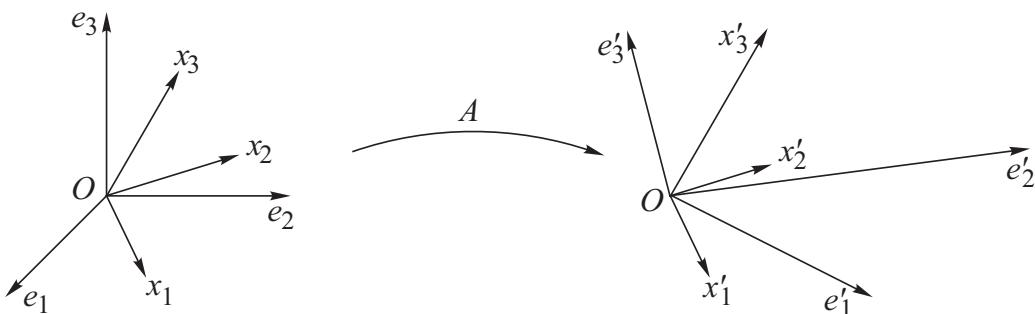


Рис. 1.5. Преобразование тройки базисных ортов пространства H и собственных векторов x_i под действием линейного оператора A

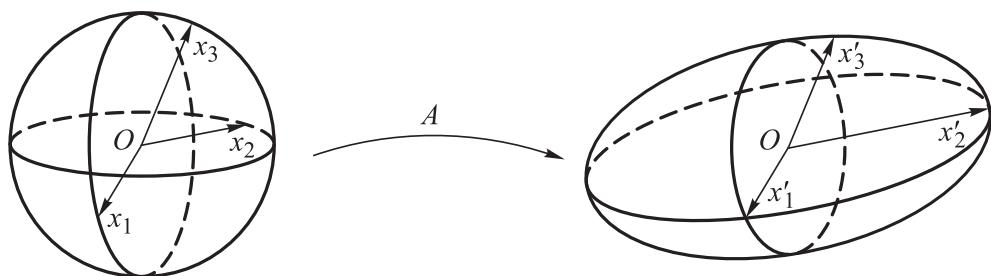


Рис. 1.6. Преобразование единичной сферы под действием самосопряженного положительно определенного оператора A

Неизменными остаются лишь направления собственных векторов x_i , $i = \overline{1, 3}$, оператора A , каждый из которых растягивается пропорционально коэффициенту, равному соответствующему собственному значению λ_i .

Рассмотрим геометрическое место точек, являющихся концами векторов единичной длины из \mathbb{R}^3 , — единичную сферу с центром в точке O . После действия оператора A эта сфера переходит в эллипсоид, оси которого параллельны собственным векторам оператора A , а значения полуосей совпадают с собственными значениями оператора (рис. 1.6).

Максимальное собственное значение оператора A совпадает со спектральным радиусом $\rho(A)$, при этом спектральный радиус получает ясное геометрическое истолкование — это радиус сферы, описанной около эллипса, полученного преобразованием единичной сферы с помощью оператора A . Если он меньше единицы, то единичная сфера после преобразования будет сжата относительно исходного состояния, если больше — растянута. При $\rho(A) < 1$ оператор A — сжимающий, следовательно, имеет, и притом единственную, неподвижную точку.

1.2.8. Признак Адамара и теоремы Гершгорина

Информация о невырожденности матрицы и характере ее спектра ценна не только сама по себе. Она играет важную вспомогательную роль во многих вычислительных задачах.

Эти свойства определяют эффективность и скорость работы численных методов, их применимость или неприменимость в конкретной задаче.

В то же время непосредственное вычисление определителя матрицы или полное определение ее спектра представляют собой подчас более сложную задачу, чем реализация того метода, для которого требуется эта информация. Приведем три теоремы, позволяющие делать выводы о свойствах матрицы, не проводя дорогостоящих вычислений.

Определение. *Матрица A называется **невырожденной**, если ее определитель не равен нулю, и **вырожденной** в противном случае.*

Решение системы линейных алгебраических уравнений (СЛАУ) с невырожденной матрицей существует и единствено, а с вырожденной матрицей может не существовать или быть неединственным. Однородная система с квадратной матрицей имеет ненулевое решение тогда и только тогда, когда матрица системы вырожденная.

Теорема 1.4 (признак Адамара невырожденности матрицы). Пусть матрица A обладает свойством строгого диагонального преобладания: $|a_{ii}| > \sum_{\substack{j=1 \\ j \neq i}}^n |a_{ij}|$, $i = \overline{1, n}$. Тогда матрица A является невырожденной.

◀ Проведем доказательство от противного. Допустим, что матрица с указанным свойством вырожденная. Тогда СЛАУ $Ax = 0$ имеет ненулевое решение $x = (x_1, \dots, x_n)$, т. е. $\sum_{j=1}^n a_{ij}x_j = 0$, $i = \overline{1, n}$.

Выберем максимальный по абсолютному значению элемент решения:

$$x_k : |x_k| = \max |x_i| > 0.$$

Тогда для k -й строки системы

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

Поэтому

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

Отсюда получим $|a_{kk}| \leq \sum_{\substack{j=1 \\ j \neq k}}^n |a_{kj}|$, что противоречит условию строгого диагонального преобладания, т.е. матрица A невырожденная. ►

В дальнейшем увидим, что многие признаки устойчивости или существования решения представляют собой по сути иные формулировки признака Адамара применительно к конкретной решаемой задаче. Так, непосредственно из признака Адамара вытекает следующая теорема.

Теорема 1.5 (первая теорема Гершгорина). Все собственные значения матрицы A лежат в объединении **кругов Гершгорина**

$$G_i = \left\{ z \in \mathbb{C}: |z - a_{ii}| \leq \sum_{\substack{j=1 \\ j \neq i}}^n |a_{ij}| \right\}, \quad i = \overline{1, n}.$$

◀ Пусть λ — собственное значение матрицы A . Тогда матрица $A - \lambda E$ вырожденная. Признак Адамара для нее несправедлив, поэтому $|\lambda - a_{ii}| \leq \sum_{\substack{j=1 \\ j \neq i}}^n |a_{ij}|$ для некоторого i . Объединяя весь спектр, получим круги G_i . ►

Вторую теорему Гершгорина приведем без доказательства.

Теорема 1.6 (вторая теорема Гершгорина). Если указанное в первой теореме Гершгорина объединение кругов распадается

на несколько связных частей, то каждая такая часть содержит столько собственных значений, сколько кругов ее составляют.

Теоремы Гершгорина дают наиболее простой, хотя и очень грубый, способ нахождения границ спектра. Наиболее действенным он оказывается тогда, когда спектр действительный.

Пример 1.8. Рассмотрим матрицу

$$A = \begin{pmatrix} 2 & 0,4 & 0,4 \\ 0,3 & 4 & 0,4 \\ 0,1 & 0,1 & 5 \end{pmatrix}.$$

Эта матрица положительно определенная:

$$\begin{aligned} (Ax, x) &= (2x_1 + 0,4x_2 + 0,4x_3)x_1 + \\ &\quad + (0,3x_1 + 4x_2 + 0,4x_3)x_2 + (0,1x_1 + 0,1x_2 + 5x_3)x_3 = \\ &= 2x_1^2 + 4x_2^2 + 5x_3^2 + 0,7x_1x_2 + 0,5x_1x_3 + 0,5x_2x_3 \geqslant \\ &\geqslant (2 - 0,35 - 0,25)x_1^2 + (4 - 0,35 - 0,25)x_2^2 + (5 - 0,25 - 0,25)x_3^2 = \\ &= 1,4x_1^2 + 3,4x_2^2 + 4,5x_3^2 \geqslant 1,4\|x\|_2^2. \end{aligned}$$

Здесь использовано неравенство $ab \geqslant \frac{1}{2}(-a^2 - b^2)$.

Из неравенства $(Ax, x) \geqslant 1,4\|x\|_2^2$ непосредственно следует, что все вещественные собственные значения матрицы A положительны. Поскольку для действительных собственного вектора x и собственного значения λ матрицы A справедливо

$$Ax = \lambda x \Rightarrow (Ax, x) = \lambda\|x\|_2^2;$$

$$(Ax, x) \geqslant 1,4\|x\|_2^2,$$

то имеем нижнюю оценку действительного спектра матрицы $\lambda \geqslant 1,4$.

Построим круги Гершгорина. Множество, которому принадлежат собственные значения матрицы, определяется неравенствами

$$|2 - \lambda| \leq 0,8; \quad |4 - \lambda| \leq 0,7; \quad |5 - \lambda| \leq 0,2.$$

На комплексной плоскости круги занимают положение, указанное на рис. 1.7. Видно, что множество кругов Гершгорина распадается на три части, в каждой из которых находится ровно по одному собственному значению. Это свидетельствует, в частности, о том, что спектр матрицы A состоит только из вещественных чисел.

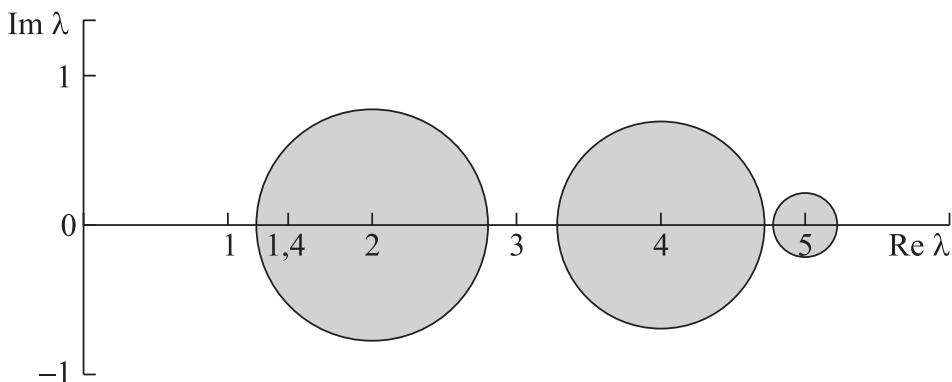


Рис. 1.7. Круги Гершгорина для матрицы из примера 1.8

Действительно, собственные числа матрицы являются корнями ее характеристического уравнения $\det(A - \lambda E) = 0$, в левой части которого стоит полином. Он имеет либо вещественные корни, либо пары комплексно-сопряженных корней с ненулевой мнимой частью. Но построенные круги лежат один за другим вдоль вещественной оси и в каждом из них находится лишь по одному корню, которые, следовательно, являются действительными. Поэтому модули могут быть раскрыты, так что собственные значения λ принадлежат множеству

$$1,2 \leq \lambda \leq 2,8; \quad 3,3 \leq \lambda \leq 4,7; \quad 4,8 \leq \lambda \leq 5,2.$$

Первый из кругов может быть подкорректирован с учетом полученной ранее оценки нижней границы спектра: $1,4 \leq \lambda \leq 2,8$. Истинные собственные значения матрицы $\lambda_1 \approx 1,9329$, $\lambda_2 \approx 4,0055$, $\lambda_3 \approx 5,0616$. •

Вопросы и задания

1. Какие способы хранения целых и действительных чисел в памяти ЭВМ вы знаете? В чем их различия?
2. Что такое абсолютная и относительная погрешности числа?
3. Как выполняется округление чисел в ЭВМ?
4. Влияют ли погрешности представления чисел в ЭВМ на результат арифметических операций и каким образом?
5. Как связаны погрешность приближенного аргумента и погрешность вычисления функции от приближенного аргумента?
6. Какая задача теории погрешностей называется прямой, а какая обратной?
7. Сформулируйте алгоритм решения обратной задачи теории погрешностей для функций многих переменных.
8. В чем различие определений нормы матрицы и нормы элемента нормированного пространства?
9. Приведите пример нормы элемента нормированного пространства квадратных матриц, не являющейся нормой матрицы.
10. Сформулируйте признак Адамара невырожденности матрицы.
11. Какие практические приложения теорем Гершгорина вы знаете? Укажите достоинства и недостатки применения теорем Гершгорина для оценки характеристик матрицы.
12. Что такое норма оператора? Какая норма называется согласованной, а какая подчиненной?

Библиографические комментарии

Материал данной главы охватывает базовые понятия, применяемые при теоретическом построении и практическом исследовании численных методов решения различных задач математического моделирования.

Базу знаний в области теоретических основ численных методов составляет функциональный анализ в конечномерных пространствах и теория линейных операторов, изложенные в работах [6] (здесь, в частности, исследована матрица Гильберта) и [10, 11, 43, 51, 53, 69]. Представляет интерес также литература по классической линейной алгебре (например, [39]), но в особенности важна специализированная литература по матричному анализу и решению задач вычислительной алгебры [17, 19, 22, 24, 34, 79, 84].

При изучении численного моделирования необходимо знать особенности представления чисел на ЭВМ. Наиболее подробно основы машинной арифметики изложены в книгах [10, 26, 27, 41, 45], ориентированных и на ручной счет. Способы работы с числами в ЭВМ детально рассмотрены также в [3, 15, 42] и др.

Во многих книгах представлены численные результаты работы алгоритмов, связанные с особенностями машинной арифметики. Например, в [7] приведены результаты вычислений интегралов из примера 1.5. Аналогичные результаты можно найти в работе [12], содержащей много примеров численной реализации алгоритмов на различных ЭВМ.

2. ПРЯМЫЕ МЕТОДЫ РЕШЕНИЯ СИСТЕМ ЛИНЕЙНЫХ АЛГЕБРАИЧЕСКИХ УРАВНЕНИЙ

Рассмотрены прямые методы решения СЛАУ. В частности, описаны алгоритмы метода Гаусса и его вариантов, включая метод квадратного корня (метод Холецкого). Представлены методы прогонки: правая, левая, встречная, циклическая, потоковая, пятидиагональная и матричная. Приведены оценки вычислительной сложности методов. Введено понятие обусловленности матрицы. Приведены основные сведения теории линейных разностных уравнений, рассмотрены методы решения разностных уравнений с постоянными коэффициентами.

2.1. Постановка задачи

Рассмотрим СЛАУ $Ax = f$, где A — невырожденная матрица размерностью $n \times n$; x — неизвестный n -мерный вектор; f — известный n -мерный вектор. В силу невырожденности матрицы A решение системы существует и единственno: $x = A^{-1}f$. Таким образом, задача решения СЛАУ, по сути, сводится к вычислению обратной матрицы A^{-1} . Отметим однако, что на практике матрица A^{-1} в явном виде почти никогда не вычисляется и не используется. Задача отыскания неизвестного вектора x более важная, а главное — вычислительно более простая.

Методы решения СЛАУ можно условно разделить на две группы: прямые и итерационные.

Прямые методы позволяют получить за конечное число действий точное решение системы уравнений при условии, что правая часть уравнений f и матрица A заданы точно,

а вычисления ведутся без погрешностей (таковы, например, методы Гаусса, квадратного корня и др.).

Итерационные методы позволяют найти приближенное решение системы путем построения последовательности приближений (итераций), сходящихся к точному решению (например, методы простой итерации, Зейделя, релаксации и др.).

Изучение методов решения СЛАУ начнем с прямых методов. Систему уравнений $Ax = f$ можно записать следующим образом:

$$\sum_{j=1}^n a_{ij} x_j = f_i, \quad i = \overline{1, n}.$$

Такая запись называется покомпонентной.

2.2. Метод Гаусса

2.2.1. Схема метода Гаусса

Идея **метода Гаусса** заключается в последовательном исключении неизвестных. Рассмотрим первое уравнение системы:

$$a_{11}x_1 + a_{12}x_2 + \dots + a_{1n}x_n = f_1.$$

Пусть $a_{11} \neq 0$. Разделим это уравнение на a_{11} . Введем следующие обозначения:

$$c_{1j} = a_{1j}/a_{11}, \quad j = \overline{2, n}; \quad y_1 = f_1/a_{11}; \quad c_{11} = 1.$$

Тогда исходную систему можно записать в виде

$$x_1 + c_{12}x_2 + \dots + c_{1n}x_n = y_1;$$

$$\sum_{j=1}^n a_{ij} x_j = f_i, \quad i = \overline{2, n}.$$

Последовательно исключим неизвестную x_1 из всех уравнений, кроме первого. Для этого умножим первое уравнение на a_{i1}

и вычтем результат из i -го уравнения ($i = \overline{2, n}$), тогда получим систему

$$\begin{aligned} x_1 + c_{12}x_2 + \dots + c_{1n}x_n &= y_1; \\ a_{22}^{(1)}x_2 + \dots + a_{2n}^{(1)}x_n &= f_2^{(1)}; \\ \dots & \\ a_{n2}^{(1)}x_2 + \dots + a_{nn}^{(1)}x_n &= f_n^{(1)}, \end{aligned}$$

где

$$a_{ij}^{(1)} = a_{ij} - a_{i1}c_{1j}, \quad f_i^{(1)} = f_i - a_{i1}y_1, \quad i, j = \overline{2, n}.$$

В результате проведенного преобразования x_1 входит только в первое уравнение, а остальные $n - 1$ уравнений можно рассматривать отдельно. Если $a_{22}^{(1)} \neq 0$, то для уравнений с номерами $i = \overline{2, n}$ можно повторить процедуру и исключить неизвестную x_2 . Повторяя эту процедуру n раз, получим систему уравнений треугольного вида:

$$\begin{aligned} x_1 + c_{12}x_2 + \dots + c_{1n}x_n &= y_1; \\ x_2 + \dots + c_{2n}x_n &= y_2; \\ \dots & \\ x_n &= y_n, \end{aligned}$$

или $Cx = y$, где C — верхняя треугольная матрица:

$$C = \begin{pmatrix} 1 & c_{12} & \dots & c_{1,n-1} & c_{1,n} \\ 0 & 1 & \dots & c_{2,n-1} & c_{2,n} \\ \dots & \dots & \dots & \dots & \dots \\ 0 & 0 & \dots & 1 & c_{n-1,n} \\ 0 & 0 & \dots & 0 & 1 \end{pmatrix}.$$

В этой матрице отличны от нуля элементы, находящиеся на главной диагонали и выше нее.

Далее выполним обратный ход:

$$x_n = y_n;$$

$$x_i = y_i - \sum_{j=i+1}^n c_{ij}x_j, \quad i = \overline{n-1, 1}.$$

Таким образом, решение СЛАУ $Ax = f$ найдено.

2.2.2. Расчетные формулы и количество действий метода Гаусса

Алгоритм решения СЛАУ методом Гаусса можно описать следующим образом:

Выполняем цикл по i от 1 до $n-1$: $y_i = f_i/a_{ii}$.

Выполняем цикл по j от $i+1$ до n : $c_{ij} = a_{ij}/a_{ii}$.

Выполняем цикл по i' от $i+1$ до n : $f_{i'} = f_{i'} - a_{i'i}y_i$.

Выполняем цикл по j' от $i+1$ до n :

$$a_{i'j'} = a_{i'j'} - a_{i'i}c_{ij'}$$

Для $i = n$ получаем $y_n = f_n/a_{nn}$.

Считаем, что значения переприсваиваются переменным в памяти ЭВМ, поэтому верхние индексы опущены.

Далее выполняем обратный ход метода Гаусса. Формулы обратного хода приведены в 2.2.1.

Главное ограничение метода Гаусса — требование отличия от нуля коэффициента $a_{ii}^{(i-1)}$, называемого **ведущим элементом** на i -м шаге исключения. На практике для того, чтобы в расчете ведущий элемент не был равен нулю или очень мал, используют модифицированный метод Гаусса с выбором ведущего элемента (см. 2.2.4).

Вычислим количество операций метода Гаусса. Ограничимся подсчетом операций умножения и деления, так как эти операции на ЭВМ занимают существенно больше времени, чем

сложение и вычитание. Для того чтобы определить количество операций, достаточно заменить каждую расчетную формулу соответствующим количеством действий, а каждый цикл — суммой с соответствующими пределами.

1. Для вычисления коэффициентов c_{ij} , $i = \overline{1, n}$; $j = \overline{i+1, n}$, необходимое количество операций равно

$$\sum_{i=1}^{n-1} \sum_{j=i+1}^n 1 = \sum_{i=1}^{n-1} (n-i) = 1 + \dots + n - 1 = \frac{1}{2}n(n-1).$$

2. Вычисление всех коэффициентов $a_{i'j'}^{(i)}$, для каждого значения i требует $(n-i)^2$ умножений (вложенный цикл по i' и j'). Для вычисления всех $a_{i'j'}^{(i)}$ необходимо выполнить следующее количество операций:

$$\begin{aligned} S_n &= \sum_{i=1}^{n-1} (n-i)^2 = 1^2 + \dots + (n-1)^2 = \sum_{i=1}^{n-1} i^2 = \\ &= \frac{1}{3} \sum_{i=1}^{n-1} [(i+1)^3 - i^3 - 3i - 1] = \\ &= \frac{1}{3} \left[n^3 - 1 - \frac{3}{2}n(n-1) - (n-1) \right] = \frac{1}{6}(n-1)n(2n-1). \end{aligned}$$

Отметим приближенный способ оценки суммы S_n . Из геометрического смысла определенного интеграла можно получить оценку

$$S_n < \int_1^n x^2 dx = \frac{1}{3}(n^3 - 1); \quad S_n > \int_1^n (x-1)^2 dx = \frac{1}{3}(n-1)^3.$$

Взяв полусумму этих оценок, получим приближенную формулу для суммы последовательных квадратов:

$$\begin{aligned} S_n &\approx \frac{1}{2} \cdot \frac{1}{3} \left[(n^3 - 1) + (n-1)^3 \right] = \frac{1}{6}(n-1)(2n^2 - n + 2) \approx \\ &\approx \frac{1}{6}(n-1)n(2n-1), \end{aligned}$$

что очень близко к точному значению S_n .

Соответственно, для вычисления элементов треугольной матрицы C требуется следующее количество операций умножения или деления:

$$\frac{1}{2}n(n-1) + \frac{1}{6}n(n-1)(2n-1) = \frac{1}{3}n(n^2-1).$$

3. Вычисление y_i требует n делений.

4. Количество операций деления при вычислении коэффициентов $f_{i'}^{(i)}$ составляет

$$\sum_{i=1}^n (n-i) = \frac{1}{2}n(n-1).$$

Итого для преобразования правой части системы линейных уравнений требуется выполнить $n(n+1)/2$ действий.

5. Для проведения прямого хода метода Гаусса необходимо выполнить следующее количество действий:

$$\frac{1}{3}n(n^2-1) + \frac{1}{2}n(n+1) = \frac{1}{6}n(n+1)(2n+1).$$

6. Для обратного хода требуется

$$\sum_{i=1}^{n-1} (n-i) = \frac{1}{2}n(n-1)$$

действий.

В результате общее количество операций деления и умножения в методе Гаусса

$$\Sigma = \frac{1}{3}n(n^2-1) + n^2 = \frac{1}{3}n(n^2+3n-1) \sim \frac{1}{3}n^3.$$

Для того чтобы понять, много или мало действий требуется в методе Гаусса, приведем пример оценки затрат машинного времени. При $n = 10^3$ следует выполнить около $1/3 \cdot 10^9$ операций. Если ЭВМ совершает 10^6 операций деления или умножения в секунду, то необходимо примерно $1/3 \cdot 10^3$ с, т. е. около 5 мин машинного времени. При $n = 10^4$ требуется уже 5000 мин, т. е. приблизительно 83 ч. Этот пример демонстрирует

ограничения на применимость метода Гаусса с точки зрения вычислительной сложности.

2.2.3. Связь метода Гаусса с разложением матрицы на множители

Представим матрицу системы уравнений $Ax = f$ в виде произведения $A = LU$, где L и U — нижняя и верхняя треугольные матрицы соответственно:

$$L = \begin{pmatrix} l_{11} & 0 & \dots & 0 & 0 \\ l_{21} & l_{22} & \dots & 0 & 0 \\ \dots & \dots & \dots & \dots & \dots \\ l_{n-1,1} & l_{n-1,2} & \cdots & l_{n-1,n-1} & 0 \\ l_{n,1} & l_{n,2} & \dots & l_{n,n-1} & l_{n,n} \end{pmatrix};$$

$$U = \begin{pmatrix} 1 & u_{12} & \cdots & u_{1,n-1} & u_{1,n} \\ 0 & 1 & \cdots & u_{2,n-1} & u_{2,n} \\ \dots & \dots & \dots & \dots & \dots \\ 0 & 0 & \cdots & 1 & u_{n-1,n} \\ 0 & 0 & \cdots & 0 & 1 \end{pmatrix}.$$

Из курса линейной алгебры известно, что для невырожденной матрицы A при некоторых дополнительных условиях это можно сделать единственным образом.

Из равенства $A = LU$ следует, что $a_{ij} = \sum_{k=1}^n l_{ik}u_{kj}$, $i, j = \overline{1, n}$.

Преобразуем эту сумму двумя способами:

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

$$\sum_{k=1}^n l_{ik}u_{kj} = \sum_{k=1}^{j-1} l_{ik}u_{kj} + l_{ij}u_{jj} + \sum_{k=j+1}^n l_{ik}u_{kj} = \sum_{k=1}^{j-1} l_{ik}u_{kj} + l_{ij}u_{jj}.$$

Поскольку диагональные элементы матрицы U равны единице ($u_{jj} = 1, j = \overline{1, n}$), из преобразованных сумм находим l_{ij} и u_{ij} :

$$l_{ij} = a_{ij} - \sum_{k=1}^{j-1} l_{ik} u_{kj} \text{ при } i > j, \quad l_{ii} = a_{ii};$$

$$u_{ij} = \frac{1}{l_{ii}} \left(a_{ij} - \sum_{k=1}^{i-1} l_{ik} u_{kj} \right) \text{ при } i < j, \quad u_{ii} = 1.$$

Последние две формулы позволяют получать матрицы L и U постепенно, по одному столбцу и одной строке соответственно. После того как матрицы L и U найдены, решение системы уравнений $Ax = f$ сводится к последовательному решению двух систем уравнений

$$Lg = f;$$

$$Ux = g$$

с матрицами L и U более простой структуры. Отметим, что матрица $U = C$ в принятых ранее обозначениях при описании метода исключения неизвестных, а процедура построения матрицы L аналогична прямому ходу метода Гаусса.

Различные разложения матрицы играют важную роль в численном анализе. Полученные множители несут важную информацию о матрице, ее ранге, структуре и свойствах; эти множители могут быть использованы не только при решении СЛАУ.

2.2.4. Выбор главного элемента

Сделаем несколько замечаний о возможностях и применимости метода Гаусса.

1. Основное ограничение применимости метода Гаусса — условие $a_{ii}^{(i-1)} \neq 0$. Исключение неизвестной x_i нельзя проводить, если в ходе расчета на главной диагонали оказался нулевой элемент. Поскольку матрица A невырождена, то в первом столбце

промежуточной системы все элементы не могут быть равны нулю. Перестановкой строк можно переместить ненулевой элемент на главную диагональ и продолжить расчет. Процедура перестановки строк соответствует изменению порядка следования уравнений в системе и, следовательно, не влияет на решение исходной системы уравнений.

Если значение $a_{ii}^{(i-1)}$ мало (по модулю), то значения c_{ij} при исключении неизвестной x_i будут велики, что в расчетах дает существенные погрешности вычисления. Поэтому на практике используют **метод Гаусса с выбором главного элемента**.

Каждую процедуру исключения неизвестной x_i начинают с поиска в i -м столбце максимального (главного) по модулю элемента $|a_{i^*i}| = \max_{i \leq i' \leq n} |a_{i'i}|$. После нахождения главного элемента строки i и i^* меняют местами и выполняют следующий шаг метода Гаусса. Перед исключением следующей неизвестной процедуру повторяют.

В ряде задач главный элемент ищут в i -й строке. В этом случае использование элемента a_{ii}^{**} приводит к перестановке столбцов, что эквивалентно перенумерации неизвестных.

С точки зрения отыскания LU -разложения перестановки строк и столбцов эквивалентны представлению матрицы в виде $A = PLU$, где P — матрица перестановок, заполненная нулями и содержащая лишь одну единицу в каждой строке и каждом столбце. Для невырожденной матрицы A такое представление всегда существует и единственno.

2. Метод Гаусса можно применять для нахождения обратной матрицы. Эта задача сводится к решению матричного уравнения $AA^{-1} = E$, где A^{-1} — неизвестная. Таким образом, процедура поиска обратной матрицы сводится к решению n систем уравнений с одной и той же матрицей A и разными правыми частями. При этом матрица A приводится к треугольному виду

лишь один раз, далее с помощью c_{ij} преобразуются только правые части уравнений и выполняется обратный ход метода Гаусса.

3. В методе Гаусса не используется информация о структуре матрицы, поэтому его можно применять для решения СЛАУ с матрицами произвольного вида. Наличие определенной структуры позволяет существенно ускорить процесс решения. По расположению ненулевых элементов различают матрицы следующих видов: ленточные, блочные, пяти- и трехдиагональные, квазитреугольные, треугольные и др. Если заранее известно расположение нулей в матрице, то вычисления можно организовать так, чтобы не включать в расчет нулевые элементы. Скорость работы при этом заметно увеличится.

4. Существуют и другие прямые методы решения СЛАУ, например методы Жордана, оптимального исключения, окаймления, отражений, ортогонализации, QR-алгоритм и др.

Кроме того, активно применяют прямые методы решения СЛАУ, использующие информацию о структуре и свойствах матрицы, такие как метод прогонки с его вариантами и метод квадратного корня.

Пример 2.1. Решим методом Гаусса СЛАУ:

$$2x_1 + 4x_2 + 3x_3 = 3;$$

$$x_1 + 2x_2 + 2x_3 = 2;$$

$$2x_1 + 4,001x_2 + 3x_3 = 2,999.$$

Выполним прямой ход метода Гаусса. Первое уравнение разделим на $a_{11} = 2$:

$$x_1 + 2x_2 + 1,5x_3 = 1,5.$$

Полученное равенство умножим на $a_{21} = 1$ и вычтем из второго уравнения, затем умножим на $a_{31} = 2$ и вычтем из третьего уравнения системы:

$$\begin{aligned} 0x_2 + 0,5x_3 &= 0,5; \\ 0,001x_2 + 0x_3 &= -0,001. \end{aligned}$$

Получена СЛАУ размерности 2, в которой $a_{22}^{(1)} = 0$. Поэтому для продолжения стандартного процесса исключения неизвестных следует поменять местами уравнения, чтобы элемент $a_{22}^{(1)}$ был отличен от нуля:

$$\begin{aligned} 0,001x_2 + 0x_3 &= -0,001; \\ 0x_2 + 0,5x_3 &= 0,5. \end{aligned}$$

В итоге система приведена к треугольному виду. В результате обратного хода метода Гаусса находим значения неизвестных: $x_3 = 1$, $x_2 = -1$, $x_1 = 2$.

На практике правые части и коэффициенты СЛАУ являются результатами вычислений и, следовательно, могут быть определены с погрешностью.

Немного изменим правую часть исходной системы (внесем погрешность), заменив правую часть системы $f = (3, 2, 2, 999)^T$ на вектор $\tilde{f} = (3, 2, 3)^T$. Получим

$$\begin{aligned} 2x_1 + 4x_2 + 3x_3 &= 3; \\ x_1 + 2x_2 + 2x_3 &= 2; \\ 2x_1 + 4,001x_2 + 3x_3 &= 3. \end{aligned}$$

Приведем систему к треугольному виду и найдем ее решение: $\tilde{x} = (0, 0, 1)^T$.

В результате получили, что небольшое возмущение $\|f - \tilde{f}\|_\infty = 0,001$ вызвало существенное изменение решения $\|\tilde{x} - x\|_\infty = 2$, т. е. относительная погрешность правой части $\|\delta f\|_\infty / \|f\|_\infty = 0,3 \cdot 10^{-3}$ привела к относительной погрешности решения $\|\delta x\|_\infty / \|x\|_\infty = 1$. •

2.3. Обусловленность систем линейных алгебраических уравнений

Определение. *Система линейных алгебраических уравнений* $Ax = f$ называется *устойчивой по правой части*, если существует такая постоянная $M \geq 0$, что для любого возмущения (погрешности) правой части δf справедлива оценка возмущения решения δx :

$$\|\delta x\| \leq M \|\delta f\|.$$

При этом постоянная M не должна зависеть от правой части f и решения x .

Если $\det A \neq 0$, то решение системы $Ax = f$ единственное и $x = A^{-1}f$. Пусть правая часть f определена с погрешностью δf . Оценим погрешность решения δx , если правая часть определена неточно. В силу линейности A и A^{-1} получим $\delta x = A^{-1}\delta f$, откуда $\|\delta x\| \leq \|A^{-1}\| \|\delta f\|$. Следовательно, $M = \|A^{-1}\|$.

Перейдем к относительным погрешностям x и f . Из оценки

$$\|f\| \leq \|A\| \|x\|$$

получим

$$\frac{\|\delta x\|}{\|x\|} \leq \frac{\|A^{-1}\|}{\|A\|^{-1}} \frac{\|\delta f\|}{\|f\|} = \|A^{-1}\| \|A\| \frac{\|\delta f\|}{\|f\|}.$$

Таким образом, если $\|A^{-1}\| \|A\|$ велико, то незначительная погрешность правой части системы уравнений может привести к существенным погрешностям в решении СЛАУ.

Определение. Число $M_A = \|A^{-1}\| \|A\|$ называется **числом обусловленности** матрицы A (и A^{-1} в силу симметрии формулы).

Матрицы с большим числом M_A называются **плохо обусловленными**. При решении СЛАУ с такими матрицами происходит резкое накопление погрешностей. При этом характеристики типа «большой» и «малый» чаще всего имеют относительный

характер и зависят как от размерности задачи, так и от возможностей (машинных и алгоритмических), которыми располагает вычислитель. Например, число обусловленности матрицы системы из примера 2.1, рассчитанное с использованием спектральной нормы матриц, $M_A \approx 2,5867 \cdot 10^4$. Отметим, что число обусловленности действительно дало верхнюю оценку коэффициента пропорциональности между относительной погрешностью правой части и относительной погрешностью решения СЛАУ.

Из оценки $\|\delta x\| \leq \|A^{-1}\| \|\delta f\|$ следует, что чем меньше определитель A , тем больше определитель A^{-1} , соответственно, больше постоянная $M = \|A^{-1}\|$ и влияние погрешностей правой части СЛАУ на погрешности решения. Обычно именно малость определителя исходной матрицы A (и большое значение определителя обратной матрицы) считают признаком плохой устойчивости решаемой системы. Однако это не всегда так, в чем убеждает простой пример матрицы $A = \varepsilon E$ с произвольным малым ε . Ее определитель очевидно мал, но проблем в решении систем с такой матрицей нет (при $\varepsilon \neq 0$), так как число обусловленности $M_A = 1$.

Пример 2.2. Рассмотрим систему уравнений

$$x_1 + 0x_2 = 1;$$

$$x_1 + 0,01x_2 = 1,$$

точное решение которой $x = (1, 0)^T$.

Внесем возмущение $\delta f = (0, 0, 01)^T$ в правую часть системы:

$$x_1 + 0x_2 = 1;$$

$$x_1 + 0,01x_2 = 1 + 0,01.$$

Решение этой системы $\tilde{x} = (1, 1)^T$.

Таким образом, в рассматриваемой системе уравнений незначительная погрешность ее правой части $\|\delta f\|_\infty / \|f\|_\infty = 0,01$

приводит к большим погрешностям решения $\|\delta x\|_\infty/\|x\|_\infty = 1$, что свидетельствует о плохой обусловленности матрицы

$$A = \begin{pmatrix} 1 & 0 \\ 1 & 0,01 \end{pmatrix}.$$

Отметим при этом, что ее число обусловленности на первый взгляд не так уж и велико: $M_A = 200,005$ при использовании спектральной нормы матриц. В то же время относительная погрешность задания коэффициентов правой части системы в 1 % привела к относительной погрешности ее решения в 100 %. Эта проблема оказывается особенно актуальной, когда правая часть системы определяется в результате измерений, для которых стандартная погрешность составляет 5 %.

Рассмотрим систему уравнений с этой же матрицей, но другой правой частью:

$$\begin{aligned} x_1 + 0x_2 &= 1; \\ x_1 + 0,01x_2 &= 1 + 10^{k-2}, \quad k \gg 1; k \in \mathbb{N}. \end{aligned}$$

Решение этой системы $\tilde{x} = (1, 10^k)^\top$. Если в ее правую часть внести погрешность $\delta f = (0, 0,01)^\top$, то решением возмущенной системы уравнений $A\tilde{x} = f + \delta f$ будет $\tilde{x} = (1, 10^k + 1)$, относительная погрешность которого $\|\delta x\|_\infty/\|x\|_\infty = 10^{-k} \rightarrow 0$ при $k \rightarrow \infty$.

Таким образом, для систем с одинаковой матрицей, но разными правыми частями одно и то же возмущение правой части может приводить как к очень значительным погрешностям решения, так и к незначительным, сравнимым с относительной погрешностью правой части СЛАУ.

Оценка погрешности решения СЛАУ через M_A представляет собой оценку сверху. При этом она оказывается точной в ряде случаев. Если решение системы $Ax = f$ близко к собственному вектору, соответствующему максимальному λ_{\max} или минимальному λ_{\min} собственному значению матрицы A ,

то решение системы наиболее чувствительно к погрешностям правой части системы. •

Теорема 2.1. Число обусловленности M_A матрицы A обладает следующими свойствами:

$$1) M_A \geq 1; \quad 2) M_A \geq \frac{|\lambda_{\max}(A)|}{|\lambda_{\min}(A)|}; \quad 3) M_{AB} \leq M_A M_B,$$

где M_{AB} — число обусловленности произведения матриц A и B .

◀ Докажем, что свойство 2 имеет место при использовании как подчиненной нормы матрицы, так и согласованной. Несложно показать, что $\|A\| \geq |\lambda_{\max}|$.

Пусть $\|A\|$ — подчиненная норма, т. е. $\|A\| = \sup_{x \neq 0} \|Ax\|/\|x\|$,

тогда $\|A\| \geq |\lambda_{\max}|$, так как в качестве вектора x под знаком супремума может стоять собственный вектор, соответствующий максимальному (по модулю) собственному значению. Для него $\|Ax\| = |\lambda_{\max}| \|x\|$.

Если $\|A\|$ — согласованная норма, то для того же вектора $Ax = \lambda_{\max}x$ из условия согласованности следует, что $\|Ax\| \leq \|A\| \|x\|$, откуда $\|A\| \geq |\lambda_{\max}|$.

Для обратной матрицы максимальным по модулю является собственное значение $|\lambda_{\min}^{-1}|$, обратное к минимальному по модулю собственному значению, откуда $\|A^{-1}\| \geq |\lambda_{\min}^{-1}|$. Следовательно,

$$M_A = \|A^{-1}\| \|A\| \geq |\lambda_{\max}| / |\lambda_{\min}|.$$

Отсюда непосредственно следует свойство 1. В свою очередь свойство 3 следует из неравенств $\|AB\| \leq \|A\| \|B\|$ для норм матриц. ►

Для пояснения геометрического смысла числа обусловленности рассмотрим совокупность X векторов, норма которых равна

единице, т. е. единичную сферу S . Среди них выделим векторы x_{\max} и x_{\min} такие, что

$$\|Ax_{\max}\| = \max_{x \in S} \|Ax\|; \quad \|Ax_{\min}\| = \min_{x \in S} \|Ax\|.$$

Очевидно, что $\|A\| = \|Ax_{\max}\|$, $\|A^{-1}\| = 1/\|Ax_{\min}\|$. Отсюда, в частности, следует, что $M_A \geq 1$.

Геометрический смысл числа обусловленности M_A наиболее нагляден, если рассматривать пространство \mathbb{R}^2 , т. е. плоскость, с евклидовой нормой $\|x\| = (x, x)^{1/2}$. В этом случае X — единичная окружность $x_1^2 + x_2^2 = 1$. При линейном преобразовании A эта окружность переходит в эллипс. Таким образом, число обусловленности M_A — отношение большой полуоси этого эллипса к его малой полуоси.

Замечание 2.1. Рассмотрим пространство H с евклидовой нормой $\|y\| = (y, y)^{1/2}$. Пусть матрица A симметрична, т. е. оператор A — самосопряженный. Тогда он имеет n различных собственных векторов x_1, x_2, \dots, x_n , образующих ортонормированный базис, и произвольный вектор x можно представить в виде

$$x = \sum_{k=1}^n c_k x_k.$$

Следовательно,

$$Ax = \sum_{k=1}^n c_k \lambda_k x_k; \quad \|Ax\|^2 = \sum_{k=1}^n c_k^2 \lambda_k^2; \quad \|x\|^2 = \sum_{k=1}^n c_k^2.$$

Отсюда $\|A\| = |\lambda_{\max}|$. Аналогично $\|A^m\| = |\lambda_{\max}^m|$, откуда $\rho(A) = |\lambda_{\max}|$. Таким же образом получаем, что $\|A^{-1}\| = |\lambda_{\min}|^{-1}$, откуда $M_A = |\lambda_{\max}|/|\lambda_{\min}|$.

Исследуем устойчивость решения СЛАУ относительно возмущений матрицы. При проведении вычислений неизбежны погрешности округления, в результате чего матрица A искается и приходится решать систему $\tilde{A}\tilde{x} = \tilde{f}$ вместо $Ax = f$.

Оценим $\delta x = \tilde{x} - x$ через $\delta A = \tilde{A} - A$. Пусть для простоты $f = \tilde{f}$. Тогда

$$(A + \delta A)(x + \delta x) = f = Ax.$$

Отсюда $(A + \delta A)\delta x = -\delta Ax$ и

$$\delta x = -(E + A^{-1}\delta A)^{-1}A^{-1}\delta Ax.$$

Следовательно,

$$\begin{aligned} \|\delta x\| &\leq \|A^{-1}\| \|\delta A\| \|(E + A^{-1}\delta A)^{-1}\| \|x\|; \\ \frac{\|\delta x\|}{\|x\|} &\leq M_A \frac{\|\delta A\|}{\|A\|} \|(E + A^{-1}\delta A)^{-1}\|. \end{aligned}$$

Будем считать последнюю норму в правой части полученного неравенства величиной порядка единицы, т. е.

$$\|(E + A^{-1}\delta A)^{-1}\| \leq C = \text{const.}$$

Приведем без доказательства оценку погрешности вычисления матрицы в методе Гаусса. Возмущение матрицы A в нем появляется вследствие представления A в ходе вычислений в виде $A = LU$, где L — нижняя треугольная матрица; U — верхняя треугольная матрица. Поэтому в результате фактически получаем $\tilde{A} = \tilde{L}\tilde{U}$. Пусть правая часть f системы задана точно. Тогда оценка возмущения матрицы A для метода Гаусса имеет вид

$$\|\delta A\|/\|A\| = O(n \cdot 2^{-t}),$$

где n — размерность матрицы; t — число разрядов мантиссы в двоичном представлении числа на ЭВМ, используемой для вычислений. Следовательно, в методе Гаусса

$$\|\delta x\|/\|x\| = O(M_A n \cdot 2^{-t}).$$

На практике найти число обусловленности конкретной матрицы, как правило, затруднительно. Тем не менее существуют приближенные итерационные методы оценки числа обусловленности, опирающиеся на построение приближенных оценок

норм матриц A и A^{-1} . Упомянем без описания метод Хейдже-ра, дающий нижнюю (но не слишком заниженную) оценку числа обусловленности за $O(n^2)$ операций при вычислении LU -разложении матрицы.

Несложно показать, что $M_A = 1$ в следующих случаях:

- 1) A — оператор подобия, т. е. $Ax = kx$;
- 2) A — ортогональный оператор, т. е. скалярное произведение $(Ax, Ax)_B \equiv (x, x)_B$;
- 3) A — произведение оператора подобия и ортогонального оператора.

Отметим, что умножение (или деление) матрицы на ненулевое число ее обусловленность не изменяет. Однако если провести эквивалентные преобразования СЛАУ (умножение на число, составление линейных комбинаций уравнений и др.), то число обусловленности системы может измениться.

Пример 2.3. Рассмотрим систему уравнений

$$\begin{aligned} \frac{1}{\varepsilon}x_1 + (1 + \varepsilon)x_2 &= f_1, \\ (1 - \varepsilon)x_1 + \varepsilon x_2 &= f_2, \end{aligned}$$

где ε — малый параметр. Оценим число обусловленности матрицы этой системы

$$A = \begin{pmatrix} 1/\varepsilon & 1 + \varepsilon \\ 1 - \varepsilon & \varepsilon \end{pmatrix}.$$

Определитель матрицы A мал: $\det A = \varepsilon^2$. Найдем собственные значения матрицы A , т. е. приравняем нулю ее определитель:

$$\begin{vmatrix} 1/\varepsilon - \lambda & 1 + \varepsilon \\ 1 - \varepsilon & \varepsilon - \lambda \end{vmatrix} = 0.$$

Получим

$$\lambda_{\max} = \frac{(1 + \varepsilon^2) + \sqrt{1 + 2\varepsilon^2 - 3\varepsilon^4}}{2\varepsilon};$$

$$\lambda_{\min} = \frac{(1 + \varepsilon^2) - \sqrt{1 + 2\varepsilon^2 - 3\varepsilon^4}}{2\varepsilon},$$

откуда оценка числа обусловленности матрицы A

$$M_A \geq \lambda_{\max}/\lambda_{\min} = O(1/\varepsilon^4).$$

В результате получили, что матрица рассматриваемой системы плохо обусловленная.

Преобразуем исходную систему уравнений, умножив первое уравнение на ε и разделив второе на ε :

$$x_1 + (1 + \varepsilon)\varepsilon x_2 = \varepsilon f_1;$$

$$\frac{1}{\varepsilon}(1 - \varepsilon)x_1 + x_2 = \frac{1}{\varepsilon}f_2.$$

Матрица преобразованной системы имеет вид

$$\tilde{A} = \begin{pmatrix} 1 & (1 + \varepsilon)\varepsilon \\ (1 - \varepsilon)/\varepsilon & 1 \end{pmatrix},$$

ее определитель $\det \tilde{A} = \varepsilon^2$. Собственные значения матрицы \tilde{A} :

$$\lambda_{\max} = 1 + \sqrt{1 - \varepsilon^2}; \quad \lambda_{\min} = 1 - \sqrt{1 - \varepsilon^2}.$$

Число обусловленности преобразованной системы

$$M_{\tilde{A}} \geq \lambda_{\max}/\lambda_{\min} = O(1/\varepsilon^2),$$

т. е. за счет эквивалентных преобразований уравнений оценка числа обусловленности улучшилась в $1/\varepsilon^2$ раз. •

Пример 2.4. Рассмотрим **матрицу Гильберта**, служащую классическим образцом плохой обусловленности. Она имеет вид $H_n = \{h_{ij}\}$, $i, j = \overline{1, n}$, с элементами $h_{ij} = (i + j - 1)^{-1}$. Норма обратной к ней матрицы экспоненциально возрастает с увеличением n . В результате число ее обусловленности при $n = 8$

превышает 10^{10} . Вследствие этого решение СЛАУ с такой матрицей может не содержать ни одной верной цифры.

Матрица Гильберта появляется естественным образом при попытке приблизить некоторый набор данных полиномом. Как будет показано далее, в главе об интерполяции функций, норма интерполяционного полинома возрастает с повышением его степени. Это свойство проявляется и в плохой обусловленности матрицы возникающей системы уравнений. •

2.4. Метод прогонки

2.4.1. Метод правой прогонки

Метод Гаусса работает значительно быстрее, если при его реализации учесть структуру матрицы системы. Рассмотрим случай трехдиагональной матрицы A , т. е. СЛАУ вида

$$a_i x_{i-1} - b_i x_i + c_i x_{i+1} = -d_i, \quad i = \overline{1, n}, \quad a_1 = c_n = 0.$$

Решим эту систему методом исключения Гаусса. Выразим неизвестную x_1 через x_2 и подставим во второе уравнение системы. В силу трехдиагональной структуры матрицы системы получится уравнение, связывающее между собой две неизвестные, но теперь это x_2 и x_3 . Далее процесс можно продолжить вплоть до исключения x_{n-1} через x_n при обработке предпоследнего уравнения системы.

Итогом описанных действий станет обнуление нижней диагонали исходной матрицы и приведение матрицы к верхнему треугольному виду. В результате последнее уравнение позволит определить x_n . После этого необходимо выполнить обратный ход метода Гаусса, последовательно определив все компоненты неизвестного вектора.

Описанный алгоритм можно реализовать, заложив в основу поиска решения линейную функциональную связь одной компоненты неизвестного вектора с последующей. Тем самым получим **метод прогонки**, который представляет собой специализированную реализацию метода исключения Гаусса, существенным образом использующую структуру матрицы системы.

Решение будем искать в виде

$$x_i = \alpha_{i+1} x_{i+1} + \beta_{i+1}, \quad i = \overline{1, n-1},$$

где α_{i+1} , β_{i+1} — прогоночные коэффициенты.

Для первого уравнения системы получим

$$x_1 = \frac{c_1}{b_1} x_2 + \frac{d_1}{b_1} = \alpha_2 x_2 + \beta_2,$$

где $\alpha_2 = c_1/b_1$; $\beta_2 = d_1/b_1$ ($b_1 \neq 0$).

Далее в строке с номером $i > 1$ исключим неизвестную x_{i-1} :

$$a_i(\alpha_i x_i + \beta_i) - b_i x_i + c_i x_{i+1} = -d_i.$$

Отсюда

$$x_i = \frac{c_i}{b_i - a_i \alpha_i} x_{i+1} + \frac{d_i + a_i \beta_i}{b_i - a_i \alpha_i} = \alpha_{i+1} x_{i+1} + \beta_{i+1},$$

$$\text{где } \alpha_{i+1} = \frac{c_i}{b_i - a_i \alpha_i}; \beta_{i+1} = \frac{d_i + a_i \beta_i}{b_i - a_i \alpha_i}, \quad i = \overline{2, n-1}.$$

При $i = n$ имеем

$$a_n(\alpha_n x_n + \beta_n) - b_n x_n = -d_n,$$

откуда

$$x_n = \frac{d_n + a_n \beta_n}{b_n - a_n \alpha_n} = \beta_{n+1}.$$

Окончательно получим **формулы прямого хода**

$$\begin{aligned} \alpha_2 &= c_1/b_1; & \beta_2 &= d_1/b_1; \\ \alpha_{i+1} &= c_i/(b_i - a_i \alpha_i); & \beta_{i+1} &= (d_i + a_i \beta_i)/(b_i - a_i \alpha_i), \quad i = \overline{2, n-1}, \end{aligned}$$

и *формулы обратного хода*

$$x_n = (d_n + a_n \beta_n) / (b_n - a_n \alpha_n) = \beta_{n+1};$$

$$x_i = \alpha_{i+1} x_{i+1} + \beta_{i+1}, \quad i = \overline{1, n-1}.$$

Количество операций деления и умножения при такой прогонке равно $5n - 4 \sim 5n$. В этой оценке учтены лишь старшие по параметру n составляющие количества действий. Оценка получена в предположении, что знаменатель в прогоночных коэффициентах вычисляется только один раз. Меньшее по порядку n количество действий при решении системы n уравнений обеспечить уже невозможно (сравните с $n^3/3$ действий в методе Гаусса).

Для успешной реализации алгоритма необходимо, чтобы в процессе расчета знаменатели не обращались в нуль, а коэффициент умножения в обратном ходе (по абсолютному значению) не превышал единицы, т. е. погрешности округления не накапливались при обратном ходе прогонки.

Определение. *Метод прогонки* называется **корректным**, если знаменатели в формулах для прогоночных коэффициентов не обращаются в нуль, и **устойчивым**, если все прогоночные коэффициенты $|\alpha_i| \leq 1$.

Теорема 2.2. Пусть в трехдиагональной матрице выполнено *условие диагонального преобладания*

$$|b_i| \geq |a_i| + |c_i|, \quad i = \overline{1, n},$$

в котором хотя бы для одного i выполнено строгое неравенство, $|b_1| > 0$ и $|c_i| > 0$, $i = \overline{2, n-1}$. Тогда система уравнений с такой матрицей имеет решение, которое может быть получено методом прогонки. Алгоритм прогонки в указанных условиях устойчив и корректен.

◀ Для доказательства корректности и устойчивости прогонки необходимо показать, что все $b_i - a_i \alpha_i \neq 0$, $i = \overline{1, n}$, и $|\alpha_i| \leq 1$,

$i = \overline{2, n}$. Доказательство первого факта опирается на принятые предположения о наличии диагонального преобладания, а второго — на метод математической индукции.

Поскольку $a_1 = 0$, то $0 \leq |a_2| \leq 1$. Пусть $|a_i| \leq 1$, где $i > 2$. Тогда, используя условие диагонального преобладания, получим

$$|b_i - a_i \alpha_i| \geq |b_i| - |a_i||\alpha_i| \geq |c_i| + |a_i|(1 - |\alpha_i|) \geq |c_i| > 0, \quad i = \overline{2, n-1}.$$

Отсюда

$$|\alpha_{i+1}| = |c_i| / |b_i - a_i \alpha_i| \leq 1.$$

Осталось показать, что $b_n - a_n \alpha_n \neq 0$. Единственная сложность доказательства вызвана тем, что $c_n = 0$. По условию теоремы хотя бы в одной строке есть строгое диагональное преобладание. Если оно достигается при $i = n$, то

$$|b_n - a_n \alpha_n| \geq |b_n| - |a_n||\alpha_n| > |a_n|(1 - |\alpha_n|) \geq 0.$$

Таким образом, знаменатель в выражении для x_n (см. формулы обратного хода) отличен от нуля. Если же строгое неравенство достигается в строке $i = i_0$, то при обработке данной строки в соответствии с проведенными выкладками получим $|\alpha_{i_0+1}| < 1$, откуда следует, что все α_i с большими номерами также меньше единицы (по модулю). ►

Замечание 2.2. Условия теоремы 2.2 носят достаточный характер. Отметим, что они во многом соответствуют *признаку Адамара* невырожденности матрицы. Описанный алгоритм весьма надежен, часто он хорошо работает и без выполнения условий диагонального преобладания.

Замечание 2.3. Приведенный вариант метода Гаусса иногда называют **методом правой прогонки**, так как непосредственное определение неизвестных происходит в нем справа налево.

2.4.2. Методы левой и встречных прогонок

Рассмотрим ту же СЛАУ с трехдиагональной матрицей A , что и в 2.4.1. Ее решение можно найти методом правой прогонки. Этот алгоритм работоспособен и практичен. Однако бывают ситуации, когда условия устойчивости этого алгоритма не выполнены, т. е. на практике алгоритм неустойчив. В этом случае может оказаться устойчивой левая прогонка. Рассмотрим ее подробнее.

В правой прогонке исключение неизвестных происходило слева направо: сначала x_1 , потом x_2 и т. д. Однако ничто не мешает исключать неизвестные в обратном порядке: сначала x_n , потом x_{n-1} и т. д. Основанный на такой процедуре алгоритм носит название **метода левой прогонки**.

Очевидность процедуры позволяет сразу выписать формулы левой прогонки:

$$x_{i+1} = \xi_{i+1}x_i + \eta_{i+1}, \quad i = \overline{1, n-1},$$

где ξ_{i+1} , η_{i+1} — прогоночные коэффициенты, которые вычисляются следующим образом:

$$\begin{aligned} \xi_n &= a_n/b_n; & \eta_n &= d_n/b_n \quad (b_n \neq 0); \\ \xi_i &= \frac{a_i}{b_i - c_i \xi_{i+1}}; & \eta_i &= \frac{d_i + c_i \eta_{i+1}}{b_i - c_i \xi_{i+1}}, \quad i = \overline{2, n-1}; \\ x_1 &= \frac{d_1 + c_1 \eta_2}{b_1 - c_1 \xi_2} = \eta_1. \end{aligned}$$

Далее ходом слева направо можно определить все остальные неизвестные.

Встречаются задачи, в которых требуется найти либо значение одной неизвестной x_m , либо группу подряд идущих значений компонентов неизвестного вектора, включающих x_m . В этом случае применим **метод встречной прогонки**. Заключается

он в последовательном исключении неизвестных с номерами, большими и меньшими m . Неизвестные с большими номерами исключаются методом левой прогонки, а с меньшими — методом правой прогонки. Для этого необходимо найти прогоночные коэффициенты $\alpha_i, \beta_i, i = \overline{2, m}$, и $\xi_i, \eta_i, i = \overline{m+1, n}$.

Далее в строке с номером m решаемой СЛАУ получим

$$a_m(\alpha_m x_m + \beta_m) - b_m x_m + c_m(\xi_{m+1} x_m + \eta_{m+1}) = -d_m.$$

Отсюда можно найти неизвестную x_m . Если необходимо определить компоненты вектора с номерами, меньшими или большими m , то применяем соответствующие формулы правой или левой прогонок с уже известными прогоночными коэффициентами. Отметим, что такой прием открывает возможность простейшего распараллеливания вычислений.

2.4.3. Метод потоковой прогонки

Потоковый вариант прогонки используют для решения систем с сильно меняющимися коэффициентами, в которых помимо неизвестных $x_i, i = \overline{1, n}$, необходимо найти еще и так называемые потоки $w_i = -a_i(x_i - x_{i-1}), i = \overline{2, n}$.

Рассмотрим СЛАУ с трехдиагональной матрицей A :

$$a_i x_{i-1} - b_i x_i + c_i x_{i+1} = -d_i, \quad i = \overline{1, n}, \quad a_1 = c_n = 0,$$

где $c_i = a_{i+1}$; $b_i = a_i + a_{i+1} + q_i, i = \overline{1, n-1}$.

Как правило, при вычислении потоков по приведенной формуле вычитание близких чисел ведет к большим погрешностям. Поэтому для получения устойчивых результатов применяют специальный алгоритм, так называемый метод потоковой прогонки, в котором такое действие исключено.

Запишем формулы правой прогонки для этой задачи:

$$x_i = \alpha_{i+1} x_{i+1} + \beta_{i+1}, \quad i = \overline{1, n-1},$$

где

$$\begin{aligned}\alpha_2 &= \frac{c_1}{b_1} = \frac{a_2}{a_2 + q_1}; & \beta_2 &= \frac{d_1}{b_1} = \frac{d_1}{a_2 + q_1}; \\ \alpha_{i+1} &= \frac{c_i}{b_i - a_i \alpha_i} = \frac{a_{i+1}}{a_{i+1} + a_i(1 - \alpha_i) + q_i}, & i &= \overline{2, n-1}; \\ \beta_{i+1} &= \frac{d_i + a_i \beta_i}{b_i - a_i \alpha_i} = \frac{d_i + a_i \beta_i}{a_{i+1} + a_i(1 - \alpha_i) + q_i}, & i &= \overline{2, n-1}.\end{aligned}$$

Преобразуем СЛАУ к виду

$$\begin{aligned}w_i - w_{i+1} - q_i x_i &= -d_i, & i &= \overline{1, n}; \\ w_1 &= w_{n+1} = 0.\end{aligned}$$

Из определения потока w_{i+1} выразим неизвестную x_i :

$$x_i = x_{i+1} + w_{i+1}/a_{i+1}.$$

С другой стороны, из предположения правой прогонки

$$x_i = \alpha_{i+1} x_{i+1} + \beta_{i+1}.$$

Получим выражение, связывающее x_{i+1} и поток w_{i+1} :

$$w_{i+1} + a_{i+1}(1 - \alpha_{i+1})x_{i+1} = a_{i+1}\beta_{i+1}.$$

Введем обозначения: $\tilde{\alpha}_{i+1} = a_{i+1}(1 - \alpha_{i+1})$; $\tilde{\beta}_{i+1} = a_{i+1}\beta_{i+1}$. Выразим x_i через w_i :

$$x_i = \frac{\tilde{\beta}_i - w_i}{\tilde{\alpha}_i}.$$

Подставим x_i в СЛАУ и запишем систему в переменных потока:

$$\begin{aligned}w_i - w_{i+1} - q_i \frac{\tilde{\beta}_i - w_i}{\tilde{\alpha}_i} &= -d_i, & i &= \overline{1, n}; \\ w_1 &= w_{n+1} = 0.\end{aligned}$$

Получили рекуррентную формулу, связывающую потоки w_i и w_{i+1} :

$$w_i = \frac{\tilde{\alpha}_i}{q_i + \tilde{\alpha}_i} w_{i+1} + \frac{q_i \tilde{\beta}_i - d_i \tilde{\alpha}_i}{q_i + \tilde{\alpha}_i}, \quad i = \overline{1, n}.$$

Поскольку $w_{n+1} = 0$, то по этой формуле найдем $w_n = \frac{q_n \tilde{\beta}_n - d_n \tilde{\alpha}_n}{q_n + \tilde{\alpha}_n}$ и последовательно определим w_i и x_i , $i = \overline{n-1, 1}$.

Далее получим рекуррентную формулу, связывающую неизвестные x_i и x_{i+1} . Подставим $w_i = -\tilde{\alpha}_i x_i + \tilde{\beta}_i$ в преобразованную СЛАУ:

$$-\tilde{\alpha}_i x_i + \tilde{\beta}_i + \tilde{\alpha}_{i+1} x_{i+1} - \tilde{\beta}_{i+1} - q_i x_i = -d_i, \quad i = \overline{1, n}.$$

Отсюда

$$x_i = \frac{\tilde{\alpha}_{i+1}}{q_i + \tilde{\alpha}_i} x_{i+1} + \frac{d_i + \tilde{\beta}_i - \tilde{\beta}_{i+1}}{q_i + \tilde{\alpha}_i}, \quad i = \overline{1, n}.$$

Из определения $\tilde{\alpha}_i$, $\tilde{\beta}_i$ и формул правой прогонки для вычисления прогоночных коэффициентов α_i и β_i несложно получить следующие рекуррентные соотношения для вычисления прогоночных коэффициентов $\tilde{\alpha}_i$, $\tilde{\beta}_i$:

$$\begin{aligned} \tilde{\alpha}_2 &= \frac{a_2 q_1}{a_2 + q_1}; & \tilde{\beta}_2 &= \frac{a_2 d_1}{a_2 + q_1}; \\ \tilde{\alpha}_{i+1} &= \frac{a_{i+1}(q_i + \tilde{\alpha}_i)}{a_{i+1} + q_i + \tilde{\alpha}_i}; & \tilde{\beta}_{i+1} &= \frac{a_{i+1}(\tilde{\beta}_i + d_i)}{a_{i+1} + q_i + \tilde{\alpha}_i}, \quad i = \overline{2, n-1}. \end{aligned}$$

Метод потоковой прогонки можно реализовать следующим образом.

1. Вычисляем прогоночные коэффициенты $\tilde{\alpha}_2$, $\tilde{\beta}_2$:

$$\tilde{\alpha}_2 = \frac{a_2 q_1}{a_2 + q_1}; \quad \tilde{\beta}_2 = \frac{a_2 d_1}{a_2 + q_1}.$$

2. Выполняем цикл по i от 2 до $n-1$:

$$\tilde{\alpha}_{i+1} = \frac{a_{i+1}(q_i + \tilde{\alpha}_i)}{a_{i+1} + q_i + \tilde{\alpha}_i}; \quad \tilde{\beta}_{i+1} = \frac{a_{i+1}(\tilde{\beta}_i + d_i)}{a_{i+1} + q_i + \tilde{\alpha}_i}.$$

3. Вычисляем поток w_n и неизвестную x_n :

$$w_n = \frac{q_n \tilde{\beta}_n - d_n \tilde{\alpha}_n}{q_n + \tilde{\alpha}_n}; \quad x_n = \frac{d_n \tilde{\beta}_n}{q_n + \tilde{\alpha}_n}.$$

4. Выполняем цикл по i от $n - 1$ до 1:

$$\begin{aligned} w_i &= \frac{\tilde{\alpha}_i}{q_i + \tilde{\alpha}_i} w_{i+1} + \frac{q_i \tilde{\beta}_i - d_i \tilde{\alpha}_i}{q_i + \tilde{\alpha}_i}; \\ x_i &= \frac{\tilde{\alpha}_{i+1}}{q_i + \tilde{\alpha}_i} x_{i+1} + \frac{d_i + \tilde{\beta}_i - \tilde{\beta}_{i+1}}{q_i + \tilde{\alpha}_i}. \end{aligned}$$

Для ускорения расчета общие для некоторых формул знаменатели можно вычислять один раз, общие для некоторых слагаемых множители выносить за скобку и умножение выполнять только один раз. Количество операций умножения и деления, необходимое для реализации метода потоковой прогонки, равно $10n$.

Если $a_i > 0$, $q_i > 0$, $i = \overline{1, n}$, то коэффициенты $\tilde{\alpha}_i > 0$. В формулах для вычисления w_i коэффициент $\tilde{\alpha}_i / (q_i + \tilde{\alpha}_i) \leq 1$, что обеспечивает устойчивость алгоритма при определении потоков w_i .

Из условий $\tilde{\alpha}_i > 0$, $q_i > 0$ следует, что $a_{i+1} < a_{i+1} + q_i + \tilde{\alpha}_i$. В силу рекуррентной формулы для $\tilde{\alpha}_i$ справедлива оценка $\tilde{\alpha}_{i+1} < q_i + \tilde{\alpha}_i$. Поэтому коэффициент $\tilde{\alpha}_{i+1} / (q_i + \tilde{\alpha}_i)$ в формулах для вычисления x_i всегда меньше единицы, что обеспечивает устойчивость этих вычислений.

2.4.4. Метод циклической прогонки

Циклическую прогонку применяют при решении задач с модифицированной матрицей, имеющей с формальной точки зрения пять ненулевых диагоналей. Рассматриваемая в этом случае система уравнений совпадает со СЛАУ из 2.4.1, кроме первого и последнего уравнений. Система имеет следующий вид:

$$\begin{aligned} a_1 x_n - b_1 x_1 + c_1 x_2 &= -d_1; \\ a_i x_{i-1} - b_i x_i + c_i x_{i+1} &= -d_i, \quad i = \overline{2, n-1}; \\ a_n x_{n-1} - b_n x_n + c_n x_1 &= -d_n, \end{aligned}$$

т. е. в матрице системы отличны от нуля главная и две примыкающие к ней слева и справа диагонали. Кроме того, ненулевые элементы находятся в правом верхнем и левом нижнем углах. Подобные задачи возникают, например, при конечномерной дискретизации дифференциальных задач с периодическими граничными условиями.

Прямой ход метода Гаусса приводит к появлению ненулевых значений в n -м столбце при преобразовании каждой строки. Поэтому решение будем искать в виде линейной комбинации, связывающей три неизвестных значения x_i , x_{i+1} и x_n :

$$x_i = \alpha_{i+1} x_{i+1} + \gamma_{i+1} x_n + \beta_{i+1},$$

где α_i , γ_i , β_i — прогоночные коэффициенты.

Из первого уравнения системы выразим неизвестную x_1 :

$$x_1 = \frac{c_1}{b_1} x_2 + \frac{a_1}{b_1} x_n + \frac{d_1}{b_1}.$$

Отсюда получим $\alpha_2 = c_1/b_1$, $\gamma_2 = a_1/b_1$, $\beta_2 = d_1/b_1$. Из i -го уравнения ($i \neq 1$) исключим неизвестную x_{i-1} :

$$a_i(\alpha_i x_i + \gamma_i x_n + \beta_i) - b_i x_i + c_i x_{i+1} = -d_i.$$

Отсюда

$$x_i = \frac{c_i}{b_i - a_i \alpha_i} x_{i+1} + \frac{a_i \gamma_i}{b_i - a_i \alpha_i} x_n + \frac{d_i + a_i \beta_i}{b_i - a_i \alpha_i}.$$

В результате можно записать следующие рекуррентные соотношения для вычисления прогоночных коэффициентов:

$$\alpha_{i+1} = \frac{c_i}{b_i - a_i \alpha_i}; \quad \gamma_{i+1} = \frac{a_i \gamma_i}{b_i - a_i \alpha_i}; \quad \beta_{i+1} = \frac{d_i + a_i \beta_i}{b_i - a_i \alpha_i}.$$

С помощью этих соотношений последовательно исключим неизвестные x_i , $i = \overline{1, n-2}$. После исключения x_{n-2} из $(n-1)$ -го уравнения получим соотношение, связывающее x_{n-1} и x_n :

$$x_{n-1} = \frac{c_{n-1} + a_{n-1}\gamma_{n-1}}{b_{n-1} - a_{n-1}\alpha_{n-1}} x_n + \frac{d_{n-1} + a_{n-1}\beta_{n-1}}{b_{n-1} - a_{n-1}\alpha_{n-1}},$$

или

$$x_{n-1} = \delta_n x_n + \omega_n,$$

где

$$\delta_n = \frac{c_{n-1} + a_{n-1}\gamma_{n-1}}{b_{n-1} - a_{n-1}\alpha_{n-1}}; \quad \omega_n = \frac{d_{n-1} + a_{n-1}\beta_{n-1}}{b_{n-1} - a_{n-1}\alpha_{n-1}}.$$

В n -е уравнение помимо x_{n-1} и x_n входит x_1 . Поэтому начинать обратный ход прогонки пока рано.

В силу предположения о виде искомого решения и полученной связи между x_{n-1} и x_n все неизвестные x_i , $i = \overline{n-1, 1}$, можно представить в виде

$$x_i = \delta_{i+1} x_n + \omega_{i+1},$$

где δ_{i+1} , ω_{i+1} — новые прогоночные коэффициенты. При этом полученное выше выражение x_i через x_{i+1} и x_n позволяет найти весь набор новых прогоночных коэффициентов с помощью рекуррентных формул

$$\delta_{i+1} = \alpha_{i+1} \delta_{i+2} + \gamma_{i+1};$$

$$\omega_{i+1} = \alpha_{i+1} \omega_{i+2} + \beta_{i+1}.$$

В итоге получим выражения для всех неизвестных x_i через x_n . В частности,

$$x_1 = \delta_2 x_n + \omega_2;$$

$$x_{n-1} = \delta_n x_n + \omega_n.$$

Рассмотрим n -е уравнение СЛАУ и подставим в него эти соотношения:

$$a_n x_{n-1} - b_n x_n + c_n x_1 = -d_n;$$

$$a_n(\delta_n x_n + \omega_n) - b_n x_n + c_n(\delta_2 x_n + \omega_2) = -d_n,$$

откуда

$$x_n = \frac{d_n + a_n \omega_n + c_n \omega_2}{b_n - a_n \delta_n - c_n \delta_2}.$$

Теперь с помощью обратного хода прогонки найдем все неизвестные x_i , $i = \overline{n-1, 1}$.

В итоге получим, что **метод циклической прогонки** можно реализовать следующим образом.

1. Вычисляем прогоночные коэффициенты α_2 , γ_2 , β_2 :

$$\alpha_2 = c_1/b_1; \quad \gamma_2 = a_1/b_1; \quad \beta_2 = d_1/b_1.$$

2. Выполняем цикл по i от 2 до $n-2$:

$$\alpha_{i+1} = \frac{c_i}{b_i - a_i \alpha_i}; \quad \gamma_{i+1} = \frac{a_i \gamma_i}{b_i - a_i \alpha_i}; \quad \beta_{i+1} = \frac{d_i + a_i \beta_i}{b_i - a_i \alpha_i}.$$

3. Вычисляем новые прогоночные коэффициенты δ_n и ω_n :

$$\delta_n = \frac{c_{n-1} - a_{n-1} \gamma_{n-1}}{b_{n-1} - a_{n-1} \alpha_{n-1}};$$

$$\omega_n = \frac{d_{n-1} - a_{n-1} \beta_{n-1}}{b_{n-1} - a_{n-1} \alpha_{n-1}}.$$

4. Выполняем цикл по i от $n-1$ до 1:

$$\delta_{i+1} = \alpha_{i+1} \delta_{i+2} + \gamma_{i+1};$$

$$\omega_{i+1} = \alpha_{i+1} \omega_{i+2} + \beta_{i+1}.$$

5. Вычисляем неизвестную x_n :

$$x_n = \frac{d_n + a_n \omega_n + c_n \omega_2}{b_n - a_n \delta_n - c_n \delta_2}.$$

6. Выполняем цикл по i от $n-1$ до 1:

$$x_i = \delta_{i+1} x_{i+1} + \omega_{i+1}.$$

Количество операций умножения и деления, необходимое для реализации циклической прогонки при решении системы размерности n , составляет $9n - 5$.

2.4.5. Метод пятидиагональной прогонки

Ранее были рассмотрены варианты метода прогонки, которые применяются для решения систем с трехдиагональной матрицей. Как правило, такие СЛАУ возникают при численном решении краевых задач для обыкновенных дифференциальных уравнений второго порядка. При решении краевых задач для уравнений более высокого порядка часто встречаются системы с пятидиагональными матрицами, в которых отличны от нуля элементы главной диагонали и примыкающих к ней слева и справа диагоналей (по две с каждой стороны). Рассмотрим такую систему:

$$-a_i x_{i-2} + b_i x_{i-1} - c_i x_i + d_i x_{i+1} - e_i x_{i+2} = -f_i, \quad i = \overline{1, n},$$

$$a_1 = b_1 = a_2 = e_{n-1} = d_n = e_n = 0.$$

Для решения СЛАУ используем метод Гаусса. В отличие от случая трехдиагональной матрицы (см. 2.4.1), при котором неизвестную x_i нужно было исключать только из одного $(i+1)$ -го уравнения, в случае пятидиагональной матрицы ее необходимо исключить из двух уравнений: $(i+1)$ -го и $(i+2)$ -го. Поэтому решение будем искать в виде

$$x_i = \alpha_{i+1} x_{i+1} - \gamma_{i+1} x_{i+2} + \beta_{i+1},$$

где α_{i+1} , γ_{i+1} , β_{i+1} — прогоночные коэффициенты.

Из первого уравнения системы найдем x_1 :

$$x_1 = \frac{d_1}{c_1} x_2 - \frac{e_1}{c_1} x_3 + \frac{f_1}{c_1} = \alpha_2 x_2 - \gamma_2 x_3 + \beta_2,$$

где $\alpha_2 = d_1/c_1$; $\gamma_2 = e_1/c_1$; $\beta_2 = f_1/c_1$.

Теперь из второго уравнения системы исключим неизвестную x_1 :

$$b_2(\alpha_2 x_2 - \gamma_2 x_3 + \beta_2) - c_2 x_2 + d_2 x_3 - e_2 x_4 = -f_2.$$

Отсюда

$$x_2 = \alpha_3 x_3 - \gamma_3 x_4 + \beta_3,$$

$$\text{где } \alpha_3 = \frac{d_2 - b_2 \gamma_2}{c_2 - b_2 \alpha_2}; \quad \gamma_3 = \frac{e_2}{c_2 - b_2 \alpha_2}; \quad \beta_3 = \frac{f_2 + b_2 \beta_2}{c_2 - b_2 \alpha_2}.$$

Продолжим исключение неизвестных по методу Гаусса. Из уравнений с номерами $i = \overline{3, n-2}$ исключим x_{i-2}, x_{i-1} :

$$\begin{aligned} & -a_i(\alpha_{i-1}(\alpha_i x_i - \gamma_i x_{i+1} + \beta_i) - \gamma_{i-1} x_i + \beta_{i-1}) + \\ & + b_i(\alpha_i x_i - \gamma_i x_{i+1} + \beta_i) - c_i x_i + d_i x_{i+1} - e_i x_{i+2} = -f_i. \end{aligned}$$

Тогда

$$x_i = \alpha_{i+1} x_{i+1} - \gamma_{i+1} x_{i+2} + \beta_{i+1}.$$

Здесь

$$\alpha_{i+1} = (a_i \alpha_{i-1} \gamma_i - b_i \gamma_i + d_i) / \Delta_{i+1}; \quad \gamma_{i+1} = e_i / \Delta_{i+1};$$

$$\beta_{i+1} = (f_i - \beta_i(a_i \alpha_{i-1} - b_i) - a_i \beta_{i-1}) / \Delta_{i+1},$$

$$\text{где } \Delta_{i+1} = c_i + a_i(\alpha_{i-1} \alpha_i - \gamma_{i-1}) - b_i \alpha_i.$$

Из $(n-1)$ -го уравнения СЛАУ следует, что $\gamma_n = 0$; α_n и β_n вычисляются по приведенным выше формулам при $i = n-1$.

Осталось найти x_n . Из последнего уравнения системы с помощью известных прогоночных коэффициентов исключим неизвестные x_{n-1} и x_{n-2} , в результате получим

$$x_n = \frac{f_n - \beta_n(a_n \alpha_{n-1} - b_n) - a_n \beta_{n-1}}{c_n + a_n(\alpha_{n-1} \alpha_n - \gamma_{n-1}) - b_n \alpha_n}.$$

Запишем теперь алгоритм **метода правой пятидиагональной прогонки**.

1. Вычисляем прогоночные коэффициенты $\alpha_2, \gamma_2, \beta_2, \alpha_3, \gamma_3, \beta_3$:

$$\alpha_2 = d_1 / c_1; \quad \gamma_2 = e_1 / c_1; \quad \beta_2 = f_1 / c_1;$$

$$\alpha_3 = \frac{d_2 - b_2 \gamma_2}{c_2 - b_2 \alpha_2}; \quad \gamma_3 = \frac{e_2}{c_2 - b_2 \alpha_2}; \quad \beta_3 = \frac{f_2 + b_2 \beta_2}{c_2 - b_2 \alpha_2}.$$

2. Выполняем цикл по i от 3 до $n - 1$:

$$\Delta_{i+1} = c_i + a_i(\alpha_{i-1}\alpha_i - \gamma_{i-1}) - b_i\alpha_i;$$

$$\alpha_{i+1} = (a_i\alpha_{i-1}\gamma_i - b_i\gamma_i + d_i) / \Delta_{i+1};$$

$$\gamma_{i+1} = e_i / \Delta_{i+1}; \quad \beta_{i+1} = (f_i - \beta_i(a_i\alpha_{i-1} - b_i) - a_i\beta_{i-1}) / \Delta_{i+1}.$$

3. Вычисляем неизвестную x_n :

$$x_n = \frac{f_n - \beta_n(a_n\alpha_{n-1} - b_n) - a_n\beta_{n-1}}{c_n + a_n(\alpha_{n-1}\alpha_n - \gamma_{n-1}) - b_n\alpha_n}.$$

4. Выполняем цикл по i от $n - 1$ до 1:

$$x_i = \alpha_{i+1}x_{i+1} - \gamma_{i+1}x_{i+2} + \beta_{i+1}.$$

Построенный **метод прогонки** будем называть **корректным**, если верны неравенства $\Delta_{i+1} = c_i + a_i(\alpha_{i-1}\alpha_i - \gamma_{i-1}) - b_i\alpha_i \neq 0$, $i = \overline{2, n-1}$, и $\Delta_2 = c_2 - b_2\alpha_2 \neq 0$. Назовем метод **устойчивым**, если верно $|\alpha_{i+1}| \leq 1$; $|\gamma_{i+1}| \leq 1$, $i = \overline{1, n-1}$.

Количество операций умножения и деления, необходимое для реализации пятидиагональной прогонки при решении системы размерности n , составляет $14n - 22$.

Замечание 2.4. В случае решения СЛАУ с $(2m + 1)$ -диагональной матрицей применим подобный алгоритм с набором из $m + 1$ прогоночных коэффициентов.

2.4.6. Метод матричной прогонки

Рассмотрим специальный случай решения СЛАУ вида

$$A_i X_{i-1} - B_i X_i + C_i X_{i+1} = -D_i, \quad i = \overline{1, n}, \quad A_1 = C_n = 0,$$

где X_i, D_i — m -мерные векторы; A_i, B_i, C_i — квадратные матрицы размерностью $m \times m$.

Подобные системы возникают при решении многомерных задач математической физики или при решении систем дифференциальных уравнений в одномерном случае.

Очевидно, что приведенная выше СЛАУ не является системой с трехдиагональной матрицей, если рассматривать все компоненты неизвестных. Однако ее специальная структура позволяет практически без изменений использовать алгоритм прогонки.

В соответствии с этим алгоритмом решение будем искать в виде

$$X_i = \alpha_{i+1} X_{i+1} + \beta_{i+1}, \quad i = \overline{1, n-1},$$

где α_{i+1} , β_{i+1} — прогоночные коэффициенты (в данном случае α_{i+1} — квадратные матрицы размерностью $m \times m$; β_{i+1} — m -мерные векторы).

Как и ранее, из первого уравнения получим

$$\alpha_2 = B_1^{-1} C_1; \quad \beta_2 = B_1^{-1} D_1$$

(матрица B_1 невырождена). Далее после преобразования i -й строки ($i \neq 1$) имеем рекуррентные формулы

$$\alpha_{i+1} = (B_i - A_i \alpha_i)^{-1} C_i; \quad \beta_{i+1} = (B_i - A_i \alpha_i)^{-1} (D_i + A_i \beta_i), \quad i = \overline{2, n-1}.$$

Преобразование строки с номером $i = n$ дает

$$X_n = (B_n - A_n \alpha_n)^{-1} (D_n + A_n \beta_n) = \beta_{n+1}.$$

Представленный алгоритм называется **методом матричной прогонки**.

В отличие от простой прогонки в данном методе деление заменяется на обращение матриц. Поэтому описанный алгоритм довольно трудоемкий: каждое обращение матриц требует порядка m^3 действий. В связи с этим для решения задач с большим m матричную прогонку используют редко.

Приведем без доказательства условия устойчивости матричной прогонки.

Теорема 2.3. Если матрицы B_i , $i = \overline{1, n}$, — невырожденные, а A_i и C_i , $i = \overline{2, n-1}$, — ненулевые и выполнены условия

$$\|B_1^{-1}C_1\| \leq 1; \quad \|B_n^{-1}A_n\| \leq 1;$$

$$\|B_i^{-1}A_i\| + \|B_i^{-1}C_i\| \leq 1, \quad i = \overline{2, n-1},$$

причем хотя бы одно из неравенств строгое, то алгоритм матричной прогонки устойчив и корректен.

2.5. Метод квадратного корня

Найдем решение СЛАУ

$$Ax = f$$

с симметричной действительной матрицей. Элементы такой матрицы удовлетворяют условию $a_{ij} = a_{ji}$, $i, j = \overline{1, n}$.

Для решения поставленной задачи существует специальная реализация метода исключения Гаусса, называемая **методом квадратного корня** или **методом Холецкого**. Метод основан на использовании структуры матрицы A , позволяющей представить матрицу в виде $A = S^*DS$, где S — верхняя треугольная матрица с положительными элементами на диагонали; S^* — ее сопряженная, т. е. нижняя треугольная, матрица; D — диагональная матрица, на диагонали которой находятся ± 1 .

По сути, метод Гаусса представляет собой разложение матрицы A в произведение $A = LU$ левой треугольной и правой треугольной матриц с последующим вычислением произведения $U^{-1}y$, где y — полученная в результате прямого хода метода правая часть системы. Для симметричной матрицы LU -разложение легко преобразовать в разложение S^*DS , на котором основан метод квадратного корня.

Пусть $S = (s_{ij})$, d_{ii} — диагональные элементы матрицы D , т. е.

$$D = \text{diag}(d_{11}, d_{22}, \dots, d_{nn}).$$

Тогда

$$(DS)_{ij} = \sum_{k=1}^n d_{ik} s_{kj} = d_{ii} s_{ij}.$$

Поскольку матрица S (как и A) действительная, то $(S^*)_{ij} = s_{ji}$ и

$$(S^* DS)_{ij} = \sum_{k=1}^n s_{ki} d_{kk} s_{kj} = a_{ij}, \quad i, j = \overline{1, n}.$$

Элемент матрицы $a_{ij} = a_{ji}$, поэтому можно рассмотреть лишь случай $i \leq j$:

$$a_{ij} = \sum_{k=1}^{i-1} s_{ki} d_{kk} s_{kj} + s_{ii} d_{ii} s_{ij} + \sum_{k=i+1}^n s_{ki} d_{kk} s_{kj}.$$

Однако $s_{ki} = 0$ при $k \geq i+1$, поэтому

$$a_{ij} = \sum_{k=1}^{i-1} s_{ki} d_{kk} s_{kj} + s_{ii} d_{ii} s_{ij}, \quad i \leq j.$$

Так, при $i = j$

$$a_{ii} = \sum_{k=1}^{i-1} d_{kk} s_{ki}^2 + s_{ii}^2 d_{ii}, \quad i = \overline{1, n}.$$

Следовательно,

$$d_{ii} = \text{sign} \left(\frac{1}{s_{ii}^2} \left(a_{ii} - \sum_{k=1}^{i-1} d_{kk} s_{ki}^2 \right) \right);$$

$$s_{ii} = \left| a_{ii} - \sum_{k=1}^{i-1} d_{kk} s_{ki}^2 \right|^{1/2}, \quad i = \overline{1, n}.$$

При $i < j$

$$s_{ij} = \frac{1}{s_{ii} d_{ii}} \left(a_{ij} - \sum_{k=1}^{i-1} s_{ki} d_{kk} s_{kj} \right).$$

Расчет осуществляется по следующему алгоритму:

$$\begin{aligned} s_{11} &= |a_{11}|^{1/2}; \quad d_{11} = \operatorname{sign}\left(\frac{a_{11}}{s_{11}^2}\right); \\ s_{1j} &= \frac{a_{1j}}{s_{11}d_{11}}, \quad j = \overline{2, n}; \\ s_{22} &= |a_{22} - d_{11}s_{12}^2|^{1/2}; \quad d_{22} = \operatorname{sign}\left(\frac{a_{22} - d_{11}s_{12}^2}{s_{22}^2}\right); \\ s_{2j} &= \frac{a_{2j} - s_{12}d_{11}s_{1j}}{s_{22}d_{22}}, \quad j = \overline{3, n}. \end{aligned}$$

Далее проводим вычисления с первым индексом $i = \overline{3, n}$.

После представления A в виде $A = S^*DS$ решение системы $Ax = f$ сводится к решению трех систем — двух треугольных и одной диагональной:

$$S^*z = f; \quad Dy = z; \quad Sx = y \quad (S^*Dy = f).$$

После исключения z имеем:

$$\begin{aligned} y_1 &= \frac{f_1}{s_{11}d_{11}}; \\ y_i &= \frac{1}{s_{ii}d_{ii}} \left(f_i - \sum_{j=1}^{i-1} s_{ji}d_{jj}y_j \right), \quad i = \overline{2, n}; \\ x_n &= \frac{y_n}{s_{nn}}; \\ x_i &= \frac{1}{s_{ii}} \left(y_i - \sum_{j=i+1}^n s_{ij}x_j \right), \quad i = \overline{n-1, 1}. \end{aligned}$$

Количество действий в методе квадратного корня примерно в 2 раза меньше количества действий в методе Гаусса за счет использования данных о структуре матрицы A , т. е. за счет ее симметрии.

2.6. Решение линейных разностных уравнений

2.6.1. Линейные разностные уравнения

Определение. Линейное уравнение относительно любых подряд идущих $m + 1$ членов последовательности $\{y_i\}_0^\infty$

$$a_0(i)y_i + a_1(i)y_{i+1} + \dots + a_m(i)y_{i+m} = f(i),$$

где $a_k(i)$, $k = \overline{0, m}$, и $f(i)$ — заданные коэффициенты, $a_0(i) \neq 0$, $a_m(i) \neq 0$, называется **линейным разностным уравнением** m -го порядка.

Решить разностное уравнение — значит найти $\{y_i\}_0^\infty$ такие, что любые расположенные подряд $m + 1$ члены последовательности удовлетворяют последнему уравнению.

Если коэффициенты a_0, a_1, \dots, a_m не зависят от i , $a_0 \neq 0$, $a_m \neq 0$, то рассматриваемое уравнение называется **линейным разностным уравнением с постоянными коэффициентами**. Если $f(i) \equiv 0$ для любого $i \geq 0$, то **разностное уравнение** называется **однородным**, в противном случае — **неоднородным**.

Простейшим примером разностных уравнений первого порядка ($m = 1$) служат формулы для членов арифметической и геометрической прогрессии:

$$y_{i+1} = y_i + d, \quad i = 0, 1, \dots;$$

$$y_{i+1} = qy_i, \quad i = 0, 1, \dots,$$

где d и q — некоторые числа.

Решение уравнения первого порядка y_i определено однозначно, если при $i = i_0$ задано условие $y_{i_0} = \tilde{y}$ (аналог задачи Коши для обыкновенного дифференциального уравнения (ОДУ)).

Еще один пример линейных разностных уравнений — СЛАУ с трехдиагональной матрицей:

$$a_i y_{i-1} - b_i y_i + c_i y_{i+1} = -d_i, \quad i = \overline{1, n}, \quad a_1 = c_n = 0.$$

Подобная задача называется краевой, поскольку ее решением является конечный набор подряд идущих y_i , $i = \overline{1, n}$. Этот набор определяется однозначно.

На практике линейные разностные уравнения часто встречаются при численном решении дифференциальных уравнений. Преобразуем рассматриваемые линейные разностные уравнения с помощью следующих формул для разностей:

$$\Delta y_i = y_{i+1} - y_i;$$

$$\Delta^2 y_i = \Delta(\Delta y_i) = (y_{i+2} - y_{i+1}) - (y_{i+1} - y_i) = y_{i+2} - 2y_{i+1} + y_i;$$

.....

$$\Delta^n y_i = \Delta(\Delta^{n-1} y_i).$$

Выразим y_{i+1} , y_{i+2} , ... через y_i и указанные разности:

$$y_{i+1} = \Delta y_i + y_i;$$

$$y_{i+2} = \Delta^2 y_i + 2y_{i+1} - y_i = \Delta^2 y_i + 2\Delta y_i + y_i \quad \text{и т. д.}$$

В результате получим новую запись разностного уравнения m -го порядка с измененными коэффициентами $\tilde{a}_i(i)$:

$$\tilde{a}_0(i)y_i + \tilde{a}_1(i)\Delta y_i + \tilde{a}_2(i)\Delta^2 y_i + \dots + \tilde{a}_m(i)\Delta^m y_i = f(i), \quad i \geq 0.$$

Данный вид уравнения объясняет, почему оно называется разностным.

Разностное уравнение m -го порядка является формальным аналогом ОДУ m -го порядка:

$$a_0(x)u + a_1(x)\frac{du}{dx} + a_2(x)\frac{d^2u}{dx^2} + \dots + a_m(x)\frac{d^mu}{dx^m} = f(x),$$

где $a_m(x) \neq 0$; $a_0(x) \neq 0$.

Рассмотрим краевую задачу для однородного разностного уравнения

$$a_0(i)y_i + a_1(i)y_{i+1} + \dots + a_m(i)y_{i+m} = 0, \quad i = \overline{0, N-m};$$

$$a_0(i) \neq 0; \quad a_m(i) \neq 0.$$

Каждое частное решение этого уравнения однозначно определяется m членами $y_{i_1}, y_{i_2}, \dots, y_{i_m}$ последовательности $\{y_i\}_0^N$. Чаще всего это заданные первые m членов решения y_0, y_1, \dots, y_{m-1} . Как и при решении линейных ОДУ, важным инструментом является понятие фундаментальной системы решений.

Теорема 2.4. Если $\{v_i^1\}_0^\infty, \{v_i^2\}_0^\infty, \dots, \{v_i^p\}_0^\infty$ — решения однородного уравнения

$$\sum_{k=0}^m a_k(i) v_{i+k} = 0,$$

то последовательность

$$\{y_i\}_0^\infty: y_i = c_1 v_i^1 + c_2 v_i^2 + \dots + c_p v_i^p,$$

где c_1, c_2, \dots, c_p — произвольные постоянные, есть решение того же однородного разностного уравнения.

◀ Подставим $y_i = \sum_{j=1}^p c_j v_i^j$ в выражение левой части однородного уравнения:

$$\sum_{k=0}^m a_k(i) y_{i+k} = \sum_{k=0}^m a_k(i) \sum_{j=1}^p c_j v_{i+k}^j.$$

Поменяем порядок суммирования в правой части равенства и воспользуемся тем, что $\{v_i^j\}_0^\infty, j = \overline{1, p}$, — решения однородного разностного уравнения:

$$\sum_{k=0}^m a_k(i) \sum_{j=1}^p c_j v_{i+k}^j = \sum_{j=1}^p c_j \sum_{k=0}^m a_k(i) v_{i+k}^j = 0.$$

Соответственно, последовательность $\{y_i\}_0^\infty$, представляющая собой линейную комбинацию частных решений $\{v_i^j\}_0^\infty, j = \overline{1, p}$, однородного уравнения, является решением этого же уравнения. ►

Решение разностного уравнения порядка m определяется константами c_i , количество которых совпадает с порядком

уравнения. Если заданы дополнительные m условий, то уравнение имеет единственное решение. Если условия заданы для последовательных m членов решения $y_i, y_{i+1}, \dots, y_{i+m-1}$, то такая задача называется **задачей Коши**. Если же дополнительные условия заданы для произвольных m членов решения $y_{i_1}, y_{i_2}, \dots, y_{i_m}$, то такая **задача** называется **краевой**.

Определение. Последовательности $\{v_i^1\}_0^\infty, \{v_i^2\}_0^\infty, \dots, \{v_i^m\}_0^\infty$ называются **линейно независимыми решениями уравнения**

$$a_0(i)y_i + a_1(i)y_{i+1} + \dots + a_m(i)y_{i+m} = f(i), \quad i \geq 0,$$

если:

- 1) $\{v_i^j\}_0^\infty, j = \overline{1, m}$, удовлетворяют этому уравнению;
- 2) соотношение $c_1 v_i^1 + c_2 v_i^2 + \dots + c_m v_i^m = 0$ при любых постоянных c_1, c_2, \dots, c_m , одновременно не равных нулю, не выполняется хотя бы для одного i .

Приведем без доказательства теоремы об общем решении линейного уравнения m -го порядка.

Теорема 2.5. Если последовательности $\{v_i^1\}_0^\infty, \{v_i^2\}_0^\infty, \dots, \{v_i^m\}_0^\infty$ — линейно независимые решения однородного уравнения m -го порядка

$$\sum_{k=0}^m a_k(i) y_{i+k} = 0,$$

то общее решение этого уравнения имеет вид

$$y_i = c_1 v_i^1 + c_2 v_i^2 + \dots + c_m v_i^m,$$

где c_1, c_2, \dots, c_m — произвольные константы. Такой набор последовательностей называется **фундаментальной системой решений линейного разностного уравнения**.

Теорема 2.6. Общее решение неоднородного линейного уравнения m -го порядка

$$\sum_{k=0}^m a_k(i) y_{i+k} = f(i)$$

можно представить в виде суммы частного решения неоднородного уравнения и общего решения линейного однородного уравнения

$$\sum_{k=0}^m a_k(i) y_{i+k} = 0.$$

Здесь также прослеживается аналогия между линейными разностными уравнениями и линейными ОДУ.

2.6.2. Линейные разностные уравнения с постоянными коэффициентами

Найдем линейно независимые решения однородного линейного уравнения с постоянными коэффициентами m -го порядка

$$a_0 y_i + a_1 y_{i+1} + \dots + a_m y_{i+m} = 0, \quad i = 0, 1, \dots .$$

Частные решения этого уравнения будем искать в виде $y_i = q^i$, где число q необходимо найти. Подставим y_i в уравнение:

$$q^i (a_0 + a_1 q + a_2 q^2 + \dots + a_m q^m) = 0, \quad i = 0, 1, \dots .$$

Поскольку ищем нетривиальное решение, то для q получим уравнение

$$a_m q^m + a_{m-1} q^{m-1} + \dots + a_1 q + a_0 = 0, \quad i = 0, 1, \dots ,$$

которое называется *характеристическим уравнением однородного линейного разностного уравнения*. Характеристическое уравнение имеет m корней, которые могут быть как простыми, так и кратными. Можно показать, что если корни q простые, то последовательности $\{y_i = q^i\}_0^\infty$, соответствующие различным

корням, линейно независимы. Поэтому общее решение однородного уравнения можно представить в виде

$$y_i = c_1 q_1^i + c_2 q_2^i + \dots + c_m q_m^i,$$

где $c_j, j = \overline{1, m}$, — произвольные постоянные.

Если корни характеристического уравнения действительны и различны, то решение y_i однозначно определяется набором коэффициентов $c_j, j = \overline{1, m}$. Аналогично обстоит дело и в случае некратных комплексных корней. Допустим, что один из корней характеристического уравнения — комплексный, т. е. $q_j = \rho(\cos \varphi + i \sin \varphi)$, $i = \sqrt{-1}$. Тогда существует сопряженный с q_j корень характеристического уравнения $q_n = \rho(\cos \varphi - i \sin \varphi)$.

Рассмотрим часть общего решения, образуемую линейной комбинацией q_j^i и q_n^i :

$$c_j q_j^i + c_n q_n^i = \rho^i [(c_j + c_n) \cos i\varphi + i(c_j - c_n) \sin i\varphi].$$

Решение y_i будет действительным, если постоянные c_j и c_n — комплексно сопряженные. Полагая, что $c_j = 0,5(\bar{c}_j - i \bar{c}_n)$ и $c_n = 0,5(\bar{c}_n + i \bar{c}_j)$, где \bar{c}_j и \bar{c}_n — произвольные действительные числа, получим $c_j q_j^i + c_n q_n^i = \rho^i (\bar{c}_j \cos i\varphi + \bar{c}_n \sin i\varphi)$.

Пусть характеристическое уравнение имеет кратные корни, т. е. существует $s \leq m$ различных корней q_i кратности σ_i , $i = \overline{1, s}$, причем $\sigma_1 + \sigma_2 + \dots + \sigma_s = m$. Тогда общее решение однородного уравнения следует искать в виде

$$y_i = \sum_{j=1}^s \sum_{n=0}^{\sigma_j-1} c_n^{(j)} i^n q_j^i,$$

где $c_n^{(j)}$ — произвольные постоянные.

Пример 2.5. Найдем решение разностного уравнения

$$-4y_{i-1} - 3y_i + y_{i+1} = 0,$$

$$y_0 = 6, \quad y_1 = 4.$$

Решение будем искать в виде $y_i = q^i$, в результате получим характеристическое уравнение

$$q^2 - 3q - 4 = 0.$$

Корни этого уравнения: $q = -1$ и $q = 4$. Поэтому общее решение однородного уравнения имеет вид $y_i = c_1 \cdot (-1)^i + c_2 \cdot 4^i$, где c_1 , c_2 — произвольные константы, значения которых определим из начальных условий:

$$y_0 = c_1 + c_2 = 6;$$

$$y_1 = c_1 \cdot (-1) + c_2 \cdot 4 = 4.$$

Решая эту систему, получим $c_1 = 4$ и $c_2 = 2$. Таким образом, решение данного разностного уравнения имеет вид

$$y_i = 4 \cdot (-1)^i + 2 \cdot 4^i. \bullet$$

Пример 2.6. Найдем общее решение однородного уравнения

$$-16y_{i-1} + 20y_i - 8y_{i+1} + y_{i+2} = 0.$$

Решение будем искать в виде $y_i = q^i$, в результате получим характеристическое уравнение

$$q^3 - 8q^2 + 20q - 16 = 0,$$

которое можно представить в виде

$$(q - 4)(q - 2)^2 = 0.$$

Характеристическое уравнение имеет два различных корня: $q_1 = 2$ и $q_2 = 4$, кратность корня q_1 равна двум, поэтому общее решение однородного уравнения имеет вид

$$y_i = c_1 \cdot 2^i + c_2 \cdot i \cdot 2^i + c_3 \cdot 4^i,$$

где c_1 , c_2 , c_3 — произвольные константы. •

Пример 2.7. Найдем общее решение разностного уравнения

$$2y_{i-1} - y_i + y_{i+1} = 0.$$

Решение представим в виде $y_i = q^i$. Составим характеристическое уравнение:

$$q^2 - q + 2 = 0.$$

Корни этого уравнения — комплексные: $q_{1,2} = (1 \pm i\sqrt{7})/2 = \sqrt{2}e^{\pm i\varphi}$, где $\varphi = \arctg\sqrt{7}$. Общее решение можно записать в виде

$$y_i = (\sqrt{2})^i (c_1 \sin i\varphi + c_2 \cos i\varphi),$$

где c_1, c_2 — произвольные константы. •

Пример 2.8. Найдем решение уравнения

$$-y_{i+1} + 2y_i = 1 + 2i - i^2.$$

Решение неоднородного уравнения — сумма общего решения линейного однородного уравнения и частного решения неоднородного уравнения.

Найдем общее решение соответствующего однородного уравнения

$$-y_{i+1} + 2y_i = 0.$$

Подставим $y_i = q^i$ и получим характеристическое уравнение $2 - q = 0$. Отсюда общее решение однородного уравнения $y_i = c \cdot 2^i$, где c — произвольная константа.

Поиск частного решения — нетривиальная задача. По аналогии с неоднородными линейными ОДУ при поиске частного решения можно исходить из вида правой части уравнения. В рассматриваемом примере в правой части стоит полином второй степени $1 + 2i - i^2$, поэтому решение будем также искать в виде полинома $\bar{y} = a_0 + a_1 i + a_2 i^2$. Подставим \bar{y} в исходное уравнение:

$$2(a_0 + a_1 i + a_2 i^2) - (a_0 + a_1(i+1) + a_2(i+1)^2) = 1 + 2i - i^2.$$

Это равенство выполняется для любого i , если коэффициенты при одинаковых степенях i^k , $k = 0, 1, 2$, в левой и правой частях

уравнения равны. Поэтому получим уравнения для искомых a_0 , a_1 и a_2 :

$$a_0 - a_1 - a_2 = 1;$$

$$a_1 - 2a_2 = 2;$$

$$a_2 = -1.$$

Отсюда $a_0 = 0$, $a_1 = 0$, $a_2 = -1$, следовательно, $\bar{y} = -i^2$. Таким образом, решение неоднородного уравнения имеет вид

$$y_i = c \cdot 2^i - i^2. \quad \bullet$$

Пример 2.9. Найдем решение уравнения

$$-6y_{i-1} - y_i + y_{i+1} = 2^{i+1}.$$

Определим общее решение соответствующего однородного уравнения. Корни характеристического уравнения $q^2 - q - 6 = 0$: $q_1 = 3$ и $q_2 = -2$. Поэтому общее решение однородного уравнения $y_i = c_1 \cdot 3^i + c_2 \cdot (-2)^i$, где c_1 , c_2 — произвольные константы.

Частное решение, исходя из вида правой части системы, будем искать в виде $\bar{y} = a \cdot 2^i$, где a — искомая константа. Подставляя \bar{y} в уравнение, получим

$$a(2^{i+1} - 2^i - 6 \cdot 2^{i-1}) = a \cdot 2^{i-1} \cdot (-4) = 2^{i+1},$$

откуда $a = -1$.

Общее решение неоднородного уравнения имеет вид

$$y_i = c_1 \cdot 3^i + c_2 \cdot (-2)^i - 2^i. \quad \bullet$$

Замечание 2.5. Как видно из примеров 2.8 и 2.9, если правая часть неоднородного уравнения $f(i)$ представляет собой многочлен, экспоненциальную функцию либо некоторую комбинацию указанных функций, вид частного решения, как правило, соответствует структуре правой части разностного уравнения, поэтому искать его лучше с помощью метода неопределенных коэффициентов.

Замечание 2.6. Как и в случае обыкновенных дифференциальных уравнений, для поиска частного решения неоднородного уравнения

$$a_0 y_i + a_1 y_{i+1} + \dots + a_m y_{i+m} = f(i)$$

можно применить метод вариации постоянных. Для этого используется общее решение однородного уравнения, в котором коэффициенты c считаются зависящими от i . Пусть $v_k(i)$, $k = \overline{1, m}$, — m линейно независимых решений однородной задачи. Тогда частное решение можно представить в виде

$$\bar{y}_i = c_1(i)v_1(i) + c_2(i)v_2(i) + \dots + c_m(i)v_m(i).$$

Введем обозначение:

$$d_k(i) = \sum_{l=1}^m [c_l(i+k) - c_l(i)] v_l(i+k), \quad k = \overline{0, m}.$$

Отметим, что $d_0 \equiv 0$. Подставим частное решение в неоднородное уравнение:

$$\begin{aligned} f(i) &= \sum_{k=0}^m a_k \bar{y}_{i+k} = \sum_{k=0}^m a_k \sum_{l=1}^m c_l(i+k) v_l(i+k) = \\ &= \sum_{k=0}^m a_k \sum_{l=1}^m [c_l(i+k) - c_l(i)] v_l(i+k) + \sum_{k=0}^m a_k \sum_{l=1}^m c_l(i) v_l(i+k) = \\ &= \sum_{k=0}^m a_k d_k(i) + \sum_{l=1}^m c_l(i) \left[\sum_{k=0}^m a_k v_l(i+k) \right]. \end{aligned}$$

Поскольку $v_k(i)$, $k = \overline{1, m}$, — решения однородного уравнения и $d_0 \equiv 0$, получим следующее соотношение:

$$\sum_{k=1}^m a_k d_k(i) = f(i),$$

которое будет выполняться, если положить, например,

$$d_k(i) = 0, \quad k = \overline{1, m-1}; \quad d_m(i) = f(i)/a_m.$$

Покажем алгоритм дальнейших действий на примере линейного разностного уравнения второго порядка. В этом случае $m = 2$ и из последней системы имеем два условия:

$$\begin{aligned} [c_1(i+1) - c_1(i)]v_1(i+1) + [c_2(i+1) - c_2(i)]v_2(i+1) &= 0; \\ [c_1(i+2) - c_1(i)]v_1(i+2) + [c_2(i+2) - c_2(i)]v_2(i+2) &= f(i)/a_2. \end{aligned}$$

Введем обозначения:

$$\begin{aligned} T_1(i) &= c_1(i+1) - c_1(i); \\ T_2(i) &= c_2(i+1) - c_2(i). \end{aligned}$$

С учетом того, что

$$\begin{aligned} c_1(i+2) - c_1(i) &= T_1(i+1) + T_1(i); \\ c_2(i+2) - c_2(i) &= T_2(i+1) + T_2(i), \end{aligned}$$

сдвинув в первом условии индекс на единицу вправо и вычтя полученное равенство из второго, можно записать следующие выражения:

$$\begin{aligned} T_1(i)v_1(i+1) + T_2(i)v_2(i+1) &= 0; \\ T_1(i)v_1(i+2) + T_2(i)v_2(i+1) &= f(i)/a_2. \end{aligned}$$

Поскольку $v_1(i)$ и $v_2(i)$ — известные функции, имеем систему двух линейных уравнений относительно величин $T_1(i)$ и $T_2(i)$. Решив эту систему, получим рекуррентные соотношения для членов последовательностей $c_1(i)$ и $c_2(i)$, полностью их определяющие.

Пример 2.10. Рассмотрим метод вариации постоянных для следующей задачи:

$$6y_i - 5y_{i+1} + y_{i+2} = 2^{i+1}.$$

Как и ранее (см. примеры 2.8 и 2.9), находим общее решение однородного уравнения: $\tilde{y}_i = c_1 \cdot 3^i + c_2 \cdot 2^i$. Решение неоднородного уравнения будем искать в виде

$$y_i = c_1(i) \cdot 3^i + c_2(i) \cdot 2^i.$$

Составим систему уравнений для определения $T_l(i) = c_l(i+1) - c_l(i)$, $l = 1, 2$:

$$\begin{aligned} T_1(i) \cdot 3^{i+1} + T_2(i) \cdot 2^{i+1} &= 0; \\ T_1(i) \cdot 3^{i+2} + T_2(i) \cdot 2^{i+2} &= 2^{i+1} \end{aligned}$$

и далее

$$\begin{aligned} T_1(i) &= 2^{i+1}/3^{i+1}; \\ T_2(i) &= -1. \end{aligned}$$

Отсюда находим

$$\begin{aligned} c_1(i+1) - c_1(i) &= (2/3)^{i+1}; \\ c_2(i+1) - c_2(i) &= -1. \end{aligned}$$

Обозначим $c_1(0) = \tilde{c}_1$, $c_2(0) = \tilde{c}_2$. Тогда (с учетом формул для суммы членов геометрической прогрессии) получим

$$\begin{aligned} c_1(i) &= \tilde{c}_1 + 2 - 2(2/3)^i; \\ c_2(i) &= \tilde{c}_2 - i. \end{aligned}$$

В итоге имеем частное решение

$$\bar{y}_i = (\tilde{c}_1 + 2)3^i + (\tilde{c}_2 - 2)2^i - i \cdot 2^i = \tilde{c}_1^* \cdot 3^i + \tilde{c}_2^* \cdot 2^i - i \cdot 2^i.$$

Решение неоднородного уравнения может быть записано в виде

$$y_i = \bar{y}_i + C_1 \cdot 3^i + C_2 \cdot 2^i = \tilde{C}_1 \cdot 3^i + \tilde{C}_2 \cdot 2^i - i \cdot 2^i. \quad \bullet$$

Замечание 2.7. Техника решения разностных уравнений может быть применена для нахождения спектра оператора. В этом случае, как правило, последовательность y_i , являющуюся нетривиальным решением разностного уравнения $A\{y_i\} = \lambda\{y_i\}$, называют собственной функцией оператора, а λ — его собственными значениями.

Пример 2.11. Найдем решение спектральной задачи

$$\begin{aligned} -\frac{1}{h^2}(\varphi_{i-1} - 2\varphi_i + \varphi_{i+1}) &= \lambda\varphi_i, \quad i = \overline{2, n-1}, \\ \varphi_1 &= \varphi_n = 0, \end{aligned}$$

где $h = \text{const}$. Требуется найти собственные значения λ и собственные функции φ_i .

Будем трактовать запись спектральной задачи как линейное разностное уравнение с постоянными коэффициентами. Решение будем искать в виде $\varphi_i = q^i$. Получим квадратное характеристическое уравнение

$$q^2 + (\lambda h^2 - 2)q + 1 = 0,$$

которое имеет два корня, причем это либо два различных корня, либо один корень кратности 2.

Рассмотрим случай кратного корня $q_1 = q_2$. По теореме Виета о корнях квадратного уравнения $q_1 q_2 = 1 \Rightarrow q_1 = q_2 = \pm 1$. Тогда решение разностного уравнения имеет вид $\varphi_i = (C_1 + C_2 i)(\pm 1)^i$. Для определения констант C_1 и C_2 подставим решение φ_i в граничные условия:

$$C_1 + C_2 = 0; \quad C_1 + C_2 n = 0.$$

Отсюда $C_1 = C_2 = 0$ и получаем $\varphi_i \equiv 0$. Это противоречит тому, что φ_i — собственная функция. Поэтому корни характеристического уравнения должны быть различными, и собственные функции φ_i следует искать в виде $\varphi_i = C_1 q_1^i + C_2 q_2^i$.

Из граничных условий спектральной задачи и теоремы Виета получаем соотношения

$$C_1 q_1 + C_2 q_2 = 0;$$

$$C_1 q_1^n + C_2 q_2^n = 0;$$

$$q_1 q_2 = 1,$$

откуда

$$q_2 = q_1^{-1};$$

$$C_1 = -C_2 q_1^2;$$

$$C_2 q_1^n \left(1 - q_1^{2(n-1)} \right) = 0.$$

Корень последнего уравнения $q_1 = 0$ не является решением спектральной задачи, остаются корни

$$q_1 = e^{\frac{i\pi k}{n-1}}, \quad k = \overline{0, 2(n-1)-1}.$$

При $k = 0$ корни $q_1 = q_2 = 1$, а при $k = n-1$ корни $q_1 = q_2 = -1$, причем в обоих случаях $\varphi_i \equiv 0$. Однако это противоречит тому, что корни характеристического уравнения различны и φ_i — собственная функция.

Собственные функции можно определить с точностью до константы:

$$\begin{aligned} \varphi_i^{(k)} &= C \left(e^{-i\frac{\pi k}{n-1}i} - e^{i\frac{\pi k}{n-1}(i-2)} \right) = \\ &= Ce^{-i\frac{\pi k}{n-1}} (-2i) \sin \frac{\pi k}{n-1}(i-1) = \tilde{C} \sin \frac{\pi k}{n-1}(i-1). \end{aligned}$$

Подставляя полученные собственные функции в исходное разностное уравнение и преобразовывая его, получим собственные значения

$$\lambda_k = \frac{4}{h^2} \sin^2 \frac{\pi k}{2(n-1)}.$$

Отметим, что при $k = n, n+1, \dots$ собственные значения и собственные функции (последние с точностью до знака) пробегают те же значения, что и при переборе $k = n-2, n-3, \dots$, поэтому система имеет $n-2$ различные собственные функции и $n-2$ различных собственных значений, которые получаются перебором $k = \overline{1, n-2}$. •

Вопросы и задания

1. Какие методы решения СЛАУ называются прямыми, а какие — итерационными? В чем их основное различие?
2. Возможно ли появление нулевого элемента на главной диагонали матрицы в процедуре исключения метода Гаусса?
3. Что такое обусловленность СЛАУ? Какая СЛАУ называется плохо обусловленной? Как связаны числа обусловленности прямой и обратной матриц?
4. Как связаны число обусловленности матрицы и определитель матрицы?
5. Как характеристики матрицы влияют на погрешности численного решения СЛАУ методом Гаусса? Чем вызвано наличие этих погрешностей?
6. Какова связь методов Гаусса и прогонки?
7. Назовите различия методов левой и правой прогонок.
8. Чем левая прогонка отличается от циклической?
9. При каком условии алгоритм матричной прогонки будет устойчивым и корректным?
10. Сформулируйте алгоритм метода Гаусса для СЛАУ с пятидиагональными матрицами.
11. Для решения каких СЛАУ применим метод квадратного корня?
12. Что такое линейное разностное уравнение? Какие уравнения называются однородными?
13. Дайте определение фундаментальной системы решений линейного разностного уравнения. Как определить ее размерность?

14. Какое уравнение называется характеристическим уравнением для линейного разностного уравнения?
15. Сформулируйте алгоритм решения неоднородного линейного разностного уравнения методом вариации постоянных.

Библиографические комментарии

Численное решение задач математической физики практически всегда приводит к необходимости решать систему линейных (или нелинейных) уравнений. Даже широко распространенное выражение «данный метод не требует решения СЛАУ» означает чаще всего лишь то, что матрица решаемой системы имеет либо диагональный вид, либо треугольный. Поиск решения таких систем не составляет проблем.

Другие случаи требуют более глубоких знаний, поэтому необходимо обратиться к соответствующей литературе. Решение систем уравнений рассматривается во всех указанных в списке литературы учебниках и учебных пособиях, например в [8, 9, 25, 35, 37, 47, 62, 65, 66].

Работы [10, 11, 69] посвящены прямым методам решения систем линейных уравнений как с плотно заполненными, так и со структурированными матрицами.

В качестве литературы, посвященной технологии работы с матрицами специального вида, укажем такие работы, как [33, 34, 63, 77, 83, 85].

Алгоритмы вычислительной линейной алгебры постоянно совершенствуются: создаются новые и модифицируются старые, классические. Например, метод Гаусса адаптируют для решения больших разреженных СЛАУ со специальными алгоритмами сжатия полосы, содержащей ненулевые элементы [30].

Современные алгоритмы вычислительной линейной алгебры представлены также в книге [24]. Развитие ЭВМ с параллельной

архитектурой привело к появлению алгоритмов, предназначенных для использования именно на таких машинах [18, 60].

Особенно интересна книга [69], в которой рассмотрено численное решение систем уравнений, возникающих при конечномерной дискретизации задач математической физики. В ней, в частности, изложена теория решения линейных разностных уравнений с постоянными и переменными коэффициентами, подробно описаны различные прямые методы решения, такие как метод редукции и метод быстрого преобразования Фурье. Указанные методы применяют для решения некоторых классов задач, количество действий при этом может достигать рекордно низких значений.

3. ИТЕРАЦИОННЫЕ МЕТОДЫ РЕШЕНИЯ СИСТЕМ ЛИНЕЙНЫХ АЛГЕБРАИЧЕСКИХ УРАВНЕНИЙ

Рассмотрены итерационные методы решения СЛАУ. Приведены стационарные и нестационарные итерационные алгоритмы, включая двух- и трехслойные. В частности, описаны методы Якоби, Зейделя, релаксации, Ричардсона с чебышевскими параметрами, а также алгоритмы вариационного типа. Введено понятие предобусловливателя. Приведен простейший алгоритм решения частичной проблемы собственных значений. Представлен один из способов регуляризации плохо обусловленных СЛАУ. Описаны способы хранения больших матриц.

3.1. Классические одношаговые итерационные методы

3.1.1. Каноническая форма одношаговых итерационных методов

Рассмотрим СЛАУ $Ax = f$, где A — невырожденная матрица размерности $n \times n$; x — неизвестный n -мерный вектор; f — известный n -мерный вектор.

Прямой метод решения этой СЛАУ может оказаться весьма ресурсозатратным в случае больших n . Кроме того, матрица A может быть слабо заполненной, но не иметь строгой структуры, что делает применение метода Гаусса бессмысленным, а использование его модификаций, типа метода прогонки, — невозможным. С другой стороны, получить при решении СЛАУ точный результат не позволяют особенности расчета на ЭВМ

(точность представления чисел, погрешности округлений) и свойства матриц (плохая обусловленность). Все это приводит к мысли о необходимости поиска приближения \tilde{x} к точному решению системы, который можно было бы осуществить быстрее, чем вычисления по методу Гаусса. При определенных условиях можно получить приближение, весьма близкое к точному решению. **Метод**, который предусматривает построение последовательности приближений $\{x^i\}$ к точному решению, будем называть **итерационным**.

Пусть решение x системы уравнений $Ax = f$ требуется найти с заданной точностью $\varepsilon > 0$, т. е. $\|\tilde{x} - x\| < \varepsilon$. Поиск решения начнем с выбора начального приближения x^0 — вектора размерности n . Затем по некоторому заданному закону выполним итерации: вычислим x^1, x^2, x^3, \dots с помощью предыдущих приближений.

Если при вычислении x^{k+1} используется значение приближенного решения только с одной предыдущей итерации x^k , то **итерационный метод** называется **одношаговым** или **двухслойным**. Если x^{k+1} вычисляется по значениям приближения с двух предыдущих итераций x^k и x^{k-1} , то метод называется **двухшаговым** или **трехслойным**, и т. д.

Любой одношаговый метод можно представить в следующей **канонической форме**:

$$B_{k+1} \frac{x^{k+1} - x^k}{\tau_{k+1}} + Ax^k = f, \quad k = 1, 2, \dots,$$

где B_{k+1} — обратимая матрица, задающая метод; k — номер итерации; x^k — k -е приближение к решению; τ_{k+1} — итерационный параметр.

Предполагается, что начальное приближение x^0 известно и произведение $B_{k+1}^{-1}r$, где r — вектор из \mathbb{R}^n , вычисляется

сравнительно просто. Тогда приближение x^{k+1} по известному x^k определяется по формуле

$$x^{k+1} = x^k + \tau_{k+1} B_{k+1}^{-1} (f - Ax^k)$$

или

$$x^{k+1} = (E - \tau_{k+1} B_{k+1}^{-1} A) x^k + \tau_{k+1} B_{k+1}^{-1} f$$

и для сходимости итерационной последовательности $\{x^k\}$ достаточно, чтобы оператор $E - \tau_{k+1} B_{k+1}^{-1} A$ был сжимающим.

Каноническая форма задает целое семейство методов. Выбор задающей метод матрицы B_{k+1} и итерационного параметра τ_{k+1} позволяет варьировать свойства итерационного процесса. Наиболее простые вычисления будут при выборе единичного оператора $B_{k+1} = E$ и постоянного параметра τ . При использовании набора $B_{k+1} = A$ и $\tau_{k+1} = 1$ точное решение будет получено за одну итерацию. Однако этот вариант совпадает с исходной задачей.

Определение. *Итерационный метод* называется **явным**, если $B_{k+1} = E$, и **неявным** в противном случае.

Отметим, что использование неявных итерационных методов оправдано лишь в том случае, когда для каждой матрицы B_k вычислить обратную матрицу проще, чем для исходной матрицы A , точнее, когда для решения системы с матрицей B_k требуется меньше машинной памяти, времени или оно алгоритмически проще, чем решение исходной системы с матрицей A . В дальнейшем запись $y = B^{-1}x$ почти никогда не будет означать получение обратной матрицы в явном виде, а будет указывать лишь на то, что y должен быть получен как решение СЛАУ $By = x$.

Определение. *Итерационный метод* называется **стационарным**, если $B_{k+1} = B$, $\tau_{k+1} = \tau$ не зависят от номера итерации k , и **нестационарным** в противном случае.

Точность итерационного метода характеризуется **погрешностью приближения** $z^k = x^k - x$. *Итерационный процесс*

называется ***сходящимся***, если $\|z^k\| \rightarrow 0$ при $k \rightarrow \infty$. Еще одной важной характеристикой итерационного метода является ***скорость сходимости***, т. е. минимальное количество итераций, гарантирующее достижение заданной точности. Скорость сходимости зависит от того, какое начальное приближение выбрано.

Расчет решения обычно ведется до выполнения одного из следующих условий:

$$\begin{aligned} \|x^{k+1} - x^k\| &\leq \varepsilon; \\ \|x^{k+1} - x^k\| &\leq \varepsilon \|x^k\| + \varepsilon_0; \\ \left\| \frac{x^{k+1} - x^k}{\|x^k\| + \varepsilon_0} \right\| &\leq \varepsilon, \end{aligned}$$

где ε характеризует точность приближенного решения, а ε_0 — техническая константа, обеспечивающая отсутствие деления на нуль или продолжение итераций в том случае, если очередное приближение $x^k = 0$.

Указанные условия прерывания итерационного процесса определяют не нормой погрешности численного решения, а нормами его изменения за одну итерацию. Иногда это может привести к неверному заключению о сходимости метода, если, например, он сходится очень медленно. В этом случае может оказаться успешным применение другого критерия:

$$\|Ax^{k+1} - f\| \leq \varepsilon,$$

который имеет дело с операторной нормой погрешности, а именно с нормой ***невязки*** — вектора

$$r^{k+1} = Ax^{k+1} - f = A(x^{k+1} - x).$$

Этот критерий часто называют критерием по невязке. Необходимо отметить, что в случае малости нормы оператора A критерий по невязке также может оказаться неприемлемым. Тем не менее

по невязке часто можно судить о значении погрешности решения или о скорости ее убывания.

Поскольку погрешность численного решения неизвестна, идеальный критерий прерывания итерационного процесса указать невозможно. Следовательно, выбор критерия чаще всего определяется искусством вычислителя. На практике обычно используют некоторый набор критериев, проверяя сходимость итерационного процесса одновременно несколькими способами.

3.1.2. Одношаговые итерационные методы

Рассмотрим несколько одношаговых итерационных методов, в частности методы простой итерации, Ричардсона с чебышёвскими параметрами, Якоби, Зейделя и релаксации.

Метод простой итерации:

$$\frac{x^{k+1} - x^k}{\tau} + Ax^k = f.$$

Он соответствует случаю, когда $B = E$, а итерационный параметр $\tau = \text{const}$.

Для этого метода особенно легко оценить количество итераций, необходимых для достижения заданного уровня ε невязки. Действительно, учитя равенство $r^k = Ax^k - f$, легко получить, что

$$A^{-1}(Ax^{k+1} - f + f - Ax^k) = \tau(f - Ax^k),$$

или

$$r^{k+1} - r^k = -\tau Ar^k$$

и, наконец,

$$r^{k+1} = (E - \tau A)r^k.$$

Отсюда следует, что невязка будет сходиться к нулю в том случае, если оператор $E - \tau A$ сжимающий, а для нормы невязки

справедлива оценка

$$\|r^{k+1}\| \leq \|E - \tau A\| \|r^k\| \leq \dots \leq \|E - \tau A\|^{k+1} \|r^0\|,$$

где $r^0 = Ax^0 - f$ может быть вычислено до начала итераций. Необходимое количество итераций n является минимальным натуральным решением неравенства

$$\|E - \tau A\|^n \|r^0\| \leq \varepsilon.$$

В качестве норм должны использоваться согласованные нормы матриц и векторов, например евклидова норма для векторов и спектральная — для матрицы.

Метод Ричардсона с чебышёвскими параметрами:

$$\frac{x^{k+1} - x^k}{\tau_{k+1}} + Ax^k = f.$$

Этот метод соответствует специальному выбору параметров τ_{k+1} таких, что норма разности $x^m - x$ (m — заранее заданный номер итерации; x — точное решение) минимальна. То, что на шаге m норма погрешности минимальна, свидетельствует об оптимальности данного метода среди всех итерационных методов с количеством шагов m . Будем выбирать евклидову норму для векторов $x^m - x$ и спектральную — для операторов.

Рассмотрим метод подробнее и найдем набор итерационных параметров, обеспечивающих его оптимальность. Пусть $z^k = x^k - x$ — погрешность k -й итерации. Поскольку $f = Ax$, то уравнение для погрешности имеет вид

$$\frac{z^{k+1} - z^k}{\tau_{k+1}} + Az^k = 0.$$

Отсюда получаем

$$z^{k+1} = (E - \tau_{k+1}A)z^k;$$

$$z^m = \prod_{k=1}^m (E - \tau_k A)z^0.$$

Выбор итерационных параметров будет оптимальным, в случае если норма оператора перехода от начальной погрешности z^0 к конечной минимальна:

$$\begin{aligned}\|z^m\| &= \min_{\tau_1, \dots, \tau_m} \left\| \prod_{k=1}^m (E - \tau_k A) z^0 \right\| \leqslant \\ &\leqslant \min_{\tau_1, \dots, \tau_m} \left\| \prod_{k=1}^m (E - \tau_k A) \right\| \|z^0\|.\end{aligned}$$

Пока не делалось никаких предположений относительно свойств оператора A . Представим окончательное решение задачи выбора итерационных параметров для простейшего случая самосопряженного оператора A , удовлетворяющего условию

$$\gamma_1 E \leq A \leq \gamma_2 E, \quad \gamma_1 > 0,$$

где γ_1, γ_2 — постоянные энергетической эквивалентности операторов A и E .

В данном случае величины γ_1, γ_2 — это границы $\lambda_{\min}, \lambda_{\max}$ спектра оператора A . В условиях самосопряженности норма оператора равна его максимальному собственному значению. В результате имеем

$$\|z^m\| \leq \min_{\tau_1, \dots, \tau_m} \max_{\gamma_1 \leq t \leq \gamma_2} \left\| \prod_{k=1}^m (1 - \tau_k t) \right\| \|z^0\|.$$

Следовательно, задача поиска набора итерационных параметров свелась к нахождению полинома $P_m(t)$ степени m , равного единице при $t = 0$ и наименее уклоняющегося от нуля, а именно:

$$\min_{\tau_1, \dots, \tau_m} \max_{\gamma_1 \leq t \leq \gamma_2} \left\| \prod_{k=1}^m (1 - \tau_k t) \right\| = \max_{\gamma_1 \leq t \leq \gamma_2} |P_m(t)| = q_m;$$

$$\|z^m\| \leq q_m \|z^0\|.$$

Решение данной задачи получено В.А. Марковым в 1892 г. Искомый полином имеет вид

$$P_m(t) = q_m T_m \left(\frac{1 - \tau_0 t}{\rho_0} \right),$$

где

$$q_m = \left(T_m \left(\frac{1}{\rho_0} \right) \right)^{-1}; \quad T_m(x) = \cos(m \arccos x), \quad |x| \leq 1;$$

$$\tau_0 = \frac{2}{\gamma_1 + \gamma_2}; \quad \rho_0 = \frac{1 - \eta}{1 + \eta} \quad \left(\eta = \frac{\gamma_1}{\gamma_2} \right).$$

Здесь $T_m(x)$ — **полином Чебышёва** первого рода степени m , а q_m может быть вычислено по формуле

$$q_m = \frac{2\rho_1^m}{1 + \rho_1^{2m}},$$

где $\rho_1 = \frac{1 - \eta^{1/2}}{1 + \eta^{1/2}}$.

Из условия совпадения корней искомого полинома и полинома Чебышёва получаем формулу для итерационных параметров:

$$\tau_k = \frac{\tau_0}{1 + \rho_0 \mu_k}, \quad k = \overline{1, m},$$

где $\mu_k = -\cos \left(\frac{2k-1}{2m}\pi \right)$ — корни полинома Чебышёва $T_m(x)$. Поэтому полученный набор итерационных параметров называется чебышёвским.

Следует отметить, что метод Ричардсона с чебышёвскими параметрами оказывается неустойчивым при выполнении действий с произвольно упорядоченным набором параметров. Для получения устойчивого алгоритма расчет необходимо выполнять при специально выбранном упорядочении данного набора.

Укажем на неочевидно решаемый вопрос о выборе количества итераций в методе Ричардсона. Данный алгоритм гарантирует, что при заданном начальном приближении x^0 на шаге m после использования m параметров τ_k , $k = \overline{1, m}$, норма погрешности z^m

будет минимальной, однако это может не обеспечить достижения заданной точности. Поэтому, используя имеющийся набор итерационных параметров τ_k , $k = \overline{1, m}$, обычно выполняют следующие m шагов и снова оценивают погрешность.

Отметим, что существуют неявные варианты метода Ричардсона и более общие варианты его обоснования.

Рассмотрим **метод Якоби**. Представим матрицу A в виде $A = A_1 + D + A_2$, где A_1 , A_2 — нижняя и верхняя треугольные матрицы, соответственно; $D = \text{diag}(a_{ii})$. Запишем решаемое уравнение в виде

$$A_1x + Dx + A_2x = f.$$

Алгоритм метода Якоби задается следующим образом:

$$x^{k+1} = D^{-1}(f - A_1x^k - A_2x^k),$$

т. е.

$$Dx^{k+1} + (A - D)x^k = f;$$

$$D(x^{k+1} - x^k) + Ax^k = f.$$

Это соответствует случаю, когда $B = D$; $\tau = 1$.

Можно описать метод Якоби иначе. Выберем в каждом уравнении исходной СЛАУ компоненту вектора неизвестных, номер которой равен номеру уравнения, и выразим ее через остальные компоненты:

$$x_1 = \left(f_1 - \sum_{i=2}^n a_{1i}x_i \right) / a_{11};$$

$$x_j = \left(f_j - \sum_{\substack{i=1 \\ i \neq j}}^n a_{ji}x_i \right) / a_{jj}, \quad j = \overline{2, n-1};$$

$$x_n = \left(f_n - \sum_{i=1}^{n-1} a_{ni} x_i \right) / a_{nn}.$$

Неизвестные компоненты решения в левой части уравнений отнесем к $(k+1)$ -й итерации, а в правой части — к k -й. Получим расчетные формулы

$$x_j^{k+1} = \left(f_j - \sum_{\substack{i=1 \\ i \neq j}}^n a_{ji} x_i^k \right) / a_{jj}, \quad j = \overline{1, n}.$$

Метод Зейделя базируется на идее метода Якоби, но использует уже найденные на данной итерации компоненты вектора неизвестных в дальнейших вычислениях. Это похоже на обратный ход метода Гаусса, поэтому данный метод часто называют также методом Гаусса — Зейделя. Алгоритм метода определен соотношениями

$$(A_1 + D)x^{k+1} + A_2 x^k = f,$$

т. е.

$$(A_1 + D)(x^{k+1} - x^k) + Ax^k = f.$$

Это соответствует случаю, когда $B = A_1 + D$; $\tau = 1$.

Расчетные формулы имеют следующий вид:

$$x_j^{k+1} = \left(f_j - \sum_{i=1}^{j-1} a_{ji} x_i^{k+1} - \sum_{i=j+1}^n a_{ji} x_i^k \right) / a_{jj}, \quad j = \overline{1, n}.$$

Идея **метода релаксации** заключается в том, чтобы увеличивать точность расчета компонент вектора неизвестных путем взвешенного усреднения приближения, получаемого по методу Зейделя, и приближения с предыдущей итерации метода. При этом уже вычисленные компоненты приближения используются для расчета последующих.

В этом случае расчетные формулы можно записать в виде

$$x_j^{k+1} = (1 - \omega)x_j^k + \omega \left(f_j - \sum_{i=1}^{j-1} a_{ji}x_i^{k+1} - \sum_{i=j+1}^n a_{ji}x_i^k \right) / a_{jj}, \quad j = \overline{1, n}.$$

Каноническая форма метода релаксации такова:

$$(D + \omega A_1) \frac{x^{k+1} - x^k}{\omega} + Ax^k = f,$$

т. е. $B = D + \omega A_1$; $\tau = \omega$.

Действительно, если каноническую форму метода представить в виде

$$(E + \omega D^{-1} A_1)x^{k+1} = ((1 - \omega)E - \omega D^{-1} A_2)x^k + \omega D^{-1}f$$

и воспользоваться покомпонентной формой записи этого равенства, получим

$$x_j^{k+1} + \omega \sum_{i=1}^{j-1} \frac{a_{ji}}{a_{jj}} x_i^{k+1} = (1 - \omega)x_j^k - \omega \sum_{i=j+1}^n \frac{a_{ji}}{a_{jj}} x_i^k + \omega \frac{f_j}{a_{jj}}, \quad j = \overline{1, n},$$

откуда сразу следуют приведенные выше расчетные формулы.

При $\omega < 1$ метод также называется методом нижней релаксации, а при $\omega > 1$ — методом верхней релаксации. Подбором ω можно существенно увеличить скорость сходимости метода, особенно в случае самосопряженной матрицы A .

Замечание 3.1. Методы Зейделя и релаксации — неявные, так как в канонической форме записи матрица B не единичная. Однако, как видно из приведенных расчетных формул методов, при вычислении x^{k+1} матрицу B^{-1} получать в явном виде не требуется. Компоненты очередного приближения могут быть найдены последовательно, причем не обязательно в порядке возрастания номера строки. Основное требование заключается в использовании уже найденных компонент приближения для расчета оставшихся.

3.1.3. Геометрическая интерпретация одношаговых стационарных итерационных методов

Решение системы двух линейных уравнений можно интерпретировать как задачу поиска точки пересечения двух прямых на плоскости. В данном случае уравнения — это уравнения прямых на плоскости, а координаты точки пересечения прямых — решение системы уравнений.

Рассмотрим решение системы с помощью *метода Якоби*. В этом случае итерационный процесс организуется следующим образом:

$$\begin{aligned} a_{11}x_1^{k+1} + a_{12}x_2^k &= f_1; \\ a_{21}x_1^k + a_{22}x_2^{k+1} &= f_2. \end{aligned}$$

На рис. 3.1, *a* приведена геометрическая интерпретация метода Якоби. Прямая I соответствует первому уравнению системы, прямая II — второму; $x^k = (x_1^k, x_2^k)$ — точки, соответствующие приближениям к решению. Отметим, что ни одно из приближений x^k не лежит на прямых I и II, т. е. ни для одного из уравнений невязка не обращается в нуль.

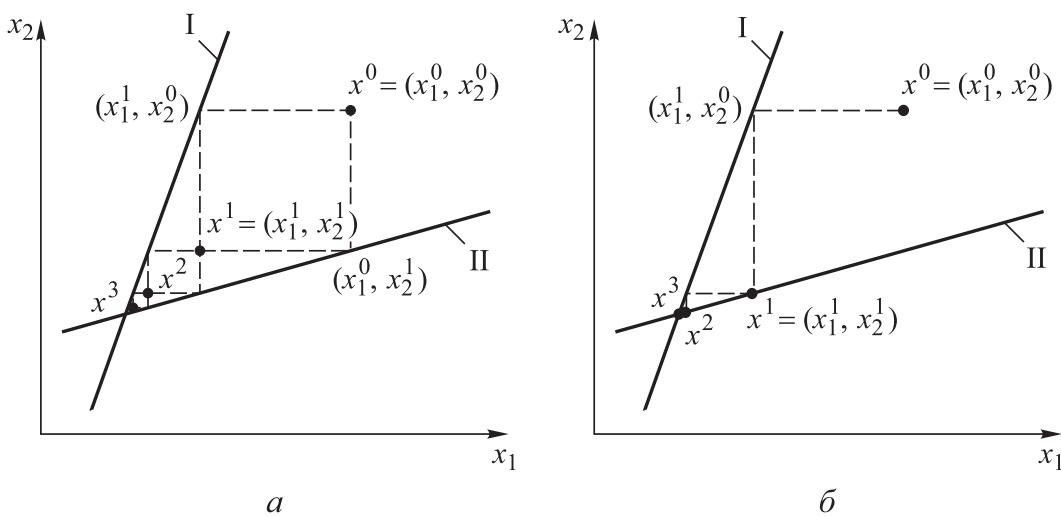


Рис. 3.1. Геометрическая интерпретация метода Якоби (*а*) и метода Зейделя (*б*)

Иная картина наблюдается при решении этой системы с помощью *метода Зейделя* (рис. 3.1, б). В этом случае итерационный процесс задается формулами

$$a_{11}x_1^{k+1} + a_{12}x_2^k = f_1;$$

$$a_{21}x_1^{k+1} + a_{22}x_2^{k+1} = f_2.$$

Здесь второе уравнение всегда решено точно.

Если изменить порядок уравнений в системе, т. е. прямой с номером I станет та, которая сейчас имеет номер II, и итерационный процесс начнется с нее, то процессы Якоби и Зейделя будут расходящимися. В этом случае норма оператора перехода от итерации к итерации становится больше единицы ($\|C\| > 1$). Поэтому перестановка уравнений в системе или перенумерация переменных в векторе неизвестных может влиять на сходимость.

При решении СЛАУ *методом релаксации* следующее приближение осуществляется по формулам

$$a_{11}(x_1^{k+1} - x_1^k) = \omega(-a_{11}x_1^k - a_{12}x_2^k + f_1);$$

$$a_{22}(x_2^{k+1} - x_2^k) = \omega(-a_{21}x_1^k - a_{22}x_2^k + f_2).$$

Таким образом, смещение вдоль оси абсцисс определяется величиной $|-a_{11}x_1^k - a_{12}x_2^k + f_1| = \omega d_I \|\vec{l}_I\|$, где d_I — расстояние от точки (x_1^k, x_2^k) до прямой I; $\vec{l}_I = (a_{11}, -a_{12})$ — направляющий вектор прямой I. При этом направление смещения вдоль оси абсцисс определяется тем, в положительной или отрицательной полуплоскости относительно прямой I расположена точка (x_1^k, x_2^k) .

Отметим, что смещение вдоль оси абсцисс

$$|x_1^{k+1} - x_1^k| = \omega \frac{d_I}{|a_{11}|/\|\vec{l}_I\|} = \omega \frac{d_I}{|\cos \beta_I|},$$

где β_I — угол между прямой I и осью ординат (рис. 3.2, а).

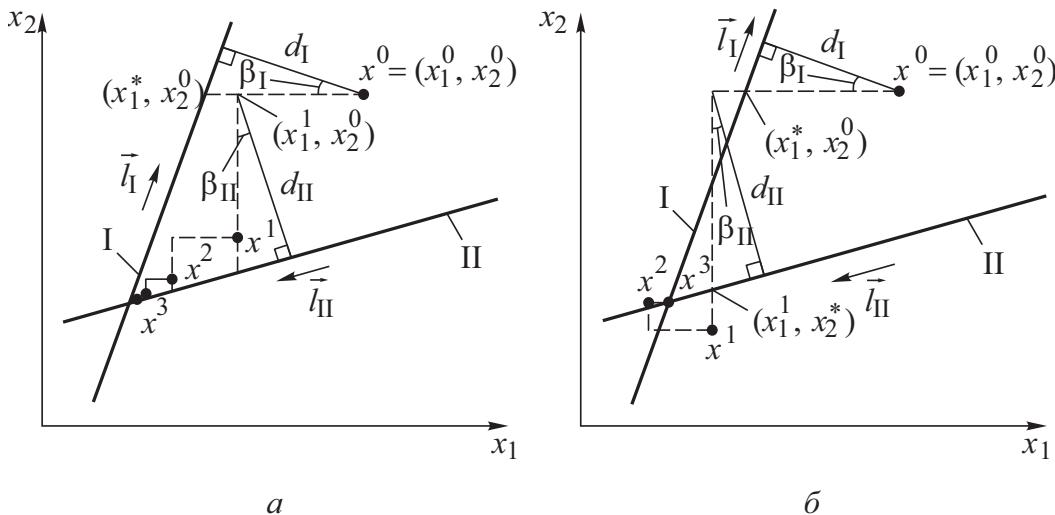


Рис. 3.2. Геометрическая интерпретация метода релаксации при $\omega < 1$ (а) и $\omega > 1$ (б)

Таким образом, $|x_1^{k+1} - x_1^k| = \omega|x_1^* - x_1^k|$, здесь x_1^* такой, что $a_{11}x_1^* + a_{12}x_2^k = f_1$. При $\omega < 1$ точки (x_1^k, x_2^k) , (x_1^{k+1}, x_2^k) находятся в одной полуплоскости относительно прямой I, а при $\omega > 1$ — в разных полуплоскостях (рис. 3.2, б).

Аналогично получим

$$|x_2^{k+1} - x_2^k| = \omega \frac{d_{II}}{|a_{22}|/\|\vec{l}_{II}\|} = \omega \frac{d_{II}}{|\cos \beta_{II}|} = \omega|x_2^* - x_2^k|,$$

где d_{II} — расстояние от точки (x_1^{k+1}, x_2^k) до прямой II; $\vec{l}_{II} = (a_{21}, -a_{22})$ — направляющий вектор прямой II; β_{II} — угол между прямой II и осью абсцисс, при этом точка (x_1^{k+1}, x_2^*) лежит на прямой II (см. рис. 3.2, б).

Прямые I и II делят плоскость на четыре сектора. При $\omega > 1$ каждое следующее приближение x^i оказывается в секторе, отличном от предыдущего приближения, а при $\omega < 1$ все приближения остаются в одном секторе. В первом случае сдвиг очередного приближения увеличивается (если направление сдвига оказалось неудачным для уменьшения невязки), а во втором —

уменьшается (если направление удачное*). При $\omega = 1$ верны равенства $x_1^{k+1} = x_1^*$, $x_2^{k+1} = x_2^*$, т. е. получаем метод Зейделя. Таким образом, подбирая параметр ω , можно увеличить скорость сходимости метода релаксации.

Если решать систему *методом простой итерации*, то вычисления проводятся по формулам

$$\frac{x^{k+1} - x^k}{\tau} + Ax^k = f,$$

которые можно записать в виде

$$x_1^{k+1} - x_1^k = \tau(-a_{11}x_1^k - a_{12}x_2^k + f_1);$$

$$x_2^{k+1} - x_2^k = \tau(-a_{21}x_1^k - a_{22}x_2^k + f_2).$$

Смещение каждого последующего приближения вдоль оси абсцисс определяется величиной $\tau |a_{11}x_1^k + a_{12}x_2^k - f_1| = \tau d_I \|\vec{l}_I\|$, где d_I — расстояние от точки (x_1^k, x_2^k) до прямой I (рис. 3.3); $\vec{l}_I = (a_{11}, -a_{12})$ — направляющий вектор прямой I. При этом

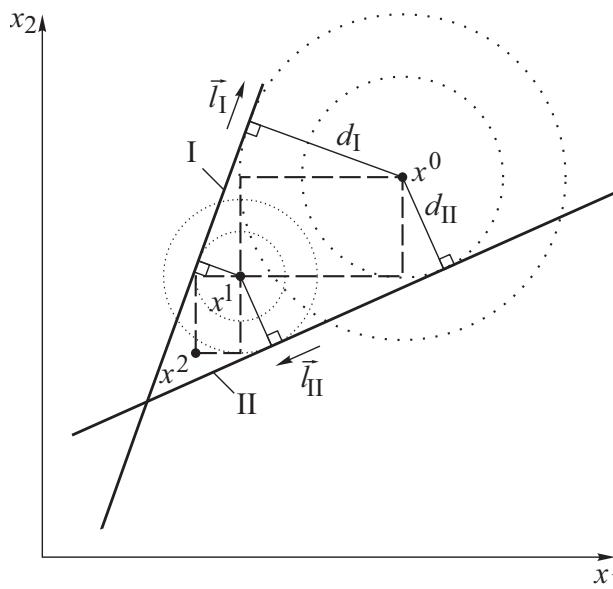


Рис. 3.3. Геометрическая интерпретация метода простой итерации при $\tau = 1$

*Часто такой метод называется SOR — Successive OverRelaxation.

направление смещения вдоль оси абсцисс определяется тем, в положительной или отрицательной полуплоскости относительно прямой I расположена точка (x_1^k, x_2^k) .

Аналогично смещение относительно оси ординат определяется величиной $\tau |a_{21}x_1^k + a_{22}x_2^k - f_2| = \tau d_{\text{II}} \|\vec{l}_{\text{II}}\|$, где d_{II} — расстояние от точки (x_1^k, x_2^k) до прямой II, $\vec{l}_{\text{II}} = (a_{21}, -a_{22})$ — направляющий вектор прямой II.

При геометрической интерпретации метода простой итерации предполагалось, что уравнения прямых нормированы ($\|\vec{l}_1\| = \|\vec{l}_{\text{II}}\| = 1$), а ориентация прямых выбрана такой, что итерационный процесс сходится.

В отличие от ранее рассмотренных методов, управлять сходимостью метода простой итерации можно за счет выбора параметра τ и выбора ориентаций прямых.

3.1.4. Условия сходимости стационарных итерационных методов

Условия сходимости стационарных итерационных методов зависят как от определяющих метод матрицы B и константы τ , так и от свойств матрицы A . Рассмотрим сначала более частный, но важный случай, допускающий относительно простую проверку условия сходимости, а затем приведем общую теорему.

Теорема 3.1. Пусть A — симметричная положительно определенная матрица, $\tau > 0$ и выполнено неравенство

$$B - 0,5\tau A > 0.$$

Тогда стационарный итерационный метод

$$B \frac{x^{k+1} - x^k}{\tau} + Ax^k = f$$

сходится при любом начальном приближении x^0 .

◀ Пусть $z^k = x^k - x$ — погрешность k -й итерации. Поскольку $f = Ax$, то

$$B \frac{z^{k+1} - z^k}{\tau} + Az^k = 0.$$

Необходимо показать, что норма погрешности стремится к нулю при $k \rightarrow \infty$. Проведем преобразования:

$$z^{k+1} = (E - \tau B^{-1} A) z^k;$$

$$Az^{k+1} = (A - \tau AB^{-1} A) z^k;$$

$$\begin{aligned} (Az^{k+1}, z^{k+1}) &= ((A - \tau AB^{-1} A) z^k, (E - \tau B^{-1} A) z^k) = \\ &= (Az^k, z^k) - \tau (Az^k, B^{-1} Az^k) - \tau (AB^{-1} Az^k, z^k) + \\ &\quad + \tau^2 (AB^{-1} Az^k, B^{-1} Az^k). \end{aligned}$$

В силу симметрии A имеем

$$(AB^{-1} Az^k, z^k) = (B^{-1} Az^k, Az^k).$$

Следовательно,

$$\begin{aligned} J_{k+1} &= (Az^{k+1}, z^{k+1}) = \\ &= J_k - 2\tau (Az^k, B^{-1} Az^k) + \tau^2 (AB^{-1} Az^k, B^{-1} Az^k) = \\ &= J_k - 2\tau ((B - 0,5\tau A) B^{-1} Az^k, B^{-1} Az^k), \end{aligned}$$

где $J_k = (Az^k, z^k)$.

Если $B - 0,5\tau A > 0$, то $J_{k+1} \leq J_k$, $J_k \geq 0$, так как $A > 0$. Отсюда заключаем, что последовательность J_k монотонно не возрастает и ограничена нулем снизу. Поэтому существует предел последовательности:

$$\lim_{k \rightarrow \infty} J_k = J.$$

Положительная определенность матрицы $B - 0,5\tau A$ означает, что верна оценка

$$((B - 0,5\tau A)y, y) \geq \delta \|y\|^2, \quad \delta > 0.$$

Следовательно,

$$J_{k+1} - J_k + 2\tau\delta \|B^{-1}Az^k\|^2 \leq 0,$$

откуда при $k \rightarrow \infty$ получим

$$\lim_{k \rightarrow \infty} \|B^{-1}Az^k\|^2 = 0.$$

Введем обозначение $\omega_k = B^{-1}Az^k$, тогда $z^k = A^{-1}B\omega_k$. Отсюда $\|z^k\| \leq \|A^{-1}B\|\|\omega_k\|$ и $\lim_{k \rightarrow \infty} \|z^k\| = 0$. ►

Следствие 3.1. Пусть A — симметричная положительно определенная матрица с диагональным преобладанием, т. е.

$$a_{ii} > \sum_{\substack{j=1 \\ j \neq i}}^n |a_{ij}|, \quad i = \overline{1, n}.$$

Тогда метод Якоби сходится.

◀ Условие сходимости в данном случае имеет вид

$$D - 0,5A > 0,$$

т. е. $A < 2D$.

Рассмотрим положительно определенную форму

$$(Ax, x) = \sum_{i,j} a_{ij} x_i x_j.$$

Для нее имеем оценку

$$\begin{aligned} (Ax, x) &\leq \frac{1}{2} \sum_{i,j} |a_{ij}| x_i^2 + \frac{1}{2} \sum_{i,j} |a_{ij}| x_j^2 = \\ &= \frac{1}{2} \sum_{i,j} |a_{ij}| x_i^2 + \frac{1}{2} \sum_{i,j} |a_{ji}| x_i^2 = \\ &= \frac{1}{2} \sum_{i,j} (|a_{ij}| + |a_{ji}|) x_i^2 = \sum_{i,j} |a_{ij}| x_i^2. \end{aligned}$$

Последнее равенство верно в силу симметричности матрицы A .

Отсюда

$$(Ax, x) \leq \sum_{i,j} |a_{ij}| x_i^2 = \sum_{i=1}^n x_i^2 \left(|a_{ii}| + \sum_{\substack{j=1 \\ j \neq i}}^n |a_{ij}| \right).$$

Однако вследствие положительной определенности матрицы $a_{ii} > 0$, $i = \overline{1, n}$. Используя условие диагонального преобладания, имеем

$$(Ax, x) < \sum_{i=1}^n x_i^2 (a_{ii} + a_{ii}) = 2(Dx, x),$$

т. е. $A < 2D$. ►

Следствие 3.2. Пусть A — симметричная положительно определенная матрица. Тогда метод релаксации сходится при $0 < \omega < 2$. В частности, сходится метод Зейделя ($\omega = 1$).

◀ Для данного метода $B = D + \omega A_1$; $\tau = \omega$; $A = A_1 + D + A_2$. В случае симметричной матрицы имеем $A_1^* = A_2$. Нужно показать, что

$$(D + \omega A_1) - 0,5\omega(A_1 + D + A_2) > 0.$$

При $0 < \omega < 2$

$$\begin{aligned} ((B - 0,5\tau A)x, x) &= \\ &= (1 - 0,5\omega)(Dx, x) + 0,5\omega(A_1 x, x) - 0,5\omega(A_2 x, x) = \\ &= (1 - 0,5\omega)(Dx, x) > 0, \end{aligned}$$

так как D — положительно определенная матрица. Случай $\omega \leq 0$ не рассматривается, так как по условию теоремы 3.1 параметр $\tau > 0$. ►

Следствие 3.3. Метод простой итерации сходится при $\tau < 2/\lambda_{\max}$, где λ_{\max} — максимальное собственное значение симметричной положительно определенной матрицы A .

◀ Условие $B - 0,5\tau A > 0$ в данном случае есть $E - 0,5\tau A > 0$, что эквивалентно условию положительности минимального собственного значения матрицы $E - 0,5\tau A$, т. е. $1 - 0,5\tau\lambda_{\max} > 0$. Отсюда $\tau < 2/\lambda_{\max}$. ►

Замечание 3.2. Безопасные на первый взгляд процедуры перестановки уравнений и перенумерации неизвестных могут нарушить диагональное преобладание и симметрию матрицы, если они были. Однако если их не было, с помощью этих же процедур можно попытаться добиться диагонального преобладания или (в редких случаях) даже симметрии матрицы.

Если матрица A не является симметричной положительно определенной, условия сходимости методов установить сложнее. В этом случае можно воспользоваться следующей теоремой (приведем ее без доказательства).

Теорема 3.2. Итерационный процесс $x^{k+1} = Cx^k + y$, где $C = E - \tau B^{-1}A$; $y = \tau B^{-1}f$, сходится к решению системы $Ax = f$, каково бы ни было начальное приближение x^0 , тогда и только тогда, когда спектральный радиус матрицы перехода между итерациями $\rho(C) < 1$, или, иными словами, когда все собственные значения матрицы C по модулю меньше единицы.

Данная теорема представляет собой частный случай принципа сжимающих отображений.

Поиск спектрального радиуса оператора — задача довольно сложная. Точные значения $\rho(C)$ удается построить лишь для специальных матриц A даже при использовании относительно простых методов, будь то метод Якоби или Зейделя. Среди таких специальных матриц, в частности, имеются и матрицы, достаточно распространенные в задачах численного решения дифференциальных уравнений.

3.2. Итерационные методы вариационного типа

3.2.1. Вариационный подход к построению итерационных методов

Описанные в предыдущем параграфе итерационные методы обычно сходятся медленно. Для применения *метода Ричардсона* необходимо знать границы спектра оператора A : минимальное λ_{\min} и максимальное λ_{\max} собственные значения. Даже при использовании *метода простой итерации* для эффективных вычислений необходимо знать λ_{\max} .

Рассмотрим итерационные методы так называемого вариационного типа решения СЛАУ, которые не требуют знания границ оператора.

Каноническая форма методов вариационного типа аналогична классическим методам:

$$B_{k+1} \frac{x^{k+1} - x^k}{\tau_{k+1}} + Ax^k = f, \quad k = 1, 2, \dots .$$

Рассмотрим каноническую форму итерационных методов с постоянными $B_{k+1} = B$ и переменными итерационными параметрами τ_{k+1} . Выберем их так, чтобы при переходе от k -й итерации к $(k+1)$ -й стала минимальной некоторая норма погрешности решения

$$\|z^{k+1}\|_D = (Dz^{k+1}, z^{k+1})^{1/2},$$

где $z^{k+1} = x^{k+1} - x$ — погрешность решения на $(k+1)$ -й итерации; D — строго положительно определенный самосопряженный линейный оператор, задающий рассматриваемый метод вместе с матрицей B .

Очевидно, что выбор $D = E$, хотя и является наиболее простым, вряд ли позволит получить оценку нормы погрешности, поскольку при неизвестном решении x не может быть вычислена ев-

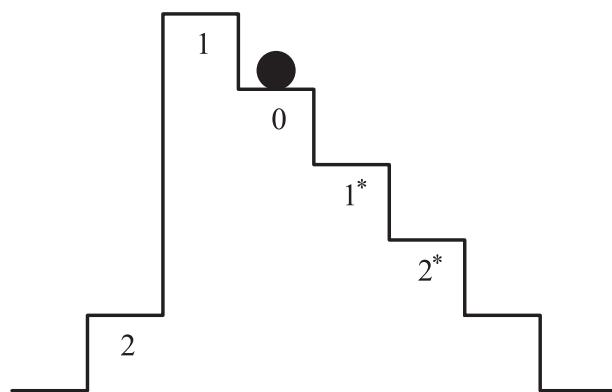


Рис. 3.4. Различные стратегии минимизации энергии шарика

клидова норма погрешности. Следует выбрать такой оператор D , который задавал бы норму погрешности, с одной стороны, легко вычисляемую, а с другой стороны, дающую не слишком искаженное представление о ее реальном значении.

Ранее был рассмотрен метод Ричардсона с чебышёвскими параметрами, обеспечивающий минимальность погрешности через заданное количество итераций m (см. 3.1.2). На первый взгляд кажется, что проведение вычислений таким образом, чтобы на каждом шаге получалась минимальная погрешность, — лучший способ нахождения решения. Однако это не так. Для пояснения приведем следующий пример: пусть шарик, находящийся в некотором начальном положении 0 (рис. 3.4), необходимо перевести в положение с минимальной энергией, т. е. переместить вниз.

Сравним результаты двух стратегий, полученные за два шага. Если требовать минимальности энергии на каждом шаге, то шарик должен двигаться по пути $0-1^*-2^*$. Если же потребовать минимальности через два шага, то получим маршрут $0-1-2$. Итог окажется заметно лучше, чем в первом варианте, что вполне объяснимо: во втором варианте шаг $0-1$ заведомо хуже с точки зрения локальной (одношаговой) оптимальности. Однако после этого шага шарик занимает очень удобную позицию, стартуя с которой он достигает минимума энергии.

3.2.2. Расчетные формулы методов вариационного типа

Перейдем в канонической форме записи итерационных методов от x^k к погрешности z^k :

$$B \frac{z^{k+1} - z^k}{\tau_{k+1}} + Az^k = 0,$$

откуда

$$z^{k+1} = (E - \tau_{k+1} B^{-1} A) z^k;$$

$$Dz^{k+1} = D(E - \tau_{k+1} B^{-1} A) z^k.$$

В результате

$$\begin{aligned} \|z^{k+1}\|_D^2 &= (Dz^k - \tau_{k+1} DB^{-1} Az^k, z^k - \tau_{k+1} B^{-1} Az^k) = \\ &= \|z^k\|_D^2 - \tau_{k+1} (DB^{-1} Az^k, z^k) - \tau_{k+1} (Dz^k, B^{-1} Az^k) + \\ &\quad + \tau_{k+1}^2 (DB^{-1} Az^k, B^{-1} Az^k). \end{aligned}$$

Поскольку D — самосопряженный оператор, то второй и третий члены в правой части полученного равенства совпадают. Квадратный относительно итерационного параметра трехчлен дает условие минимума для $\|z^{k+1}\|_D$, из которого получаем следующее реализующее минимум значение итерационного параметра:

$$\tau_{k+1} = \frac{(B^{-1} Az^k, Dz^k)}{(DB^{-1} Az^k, B^{-1} Az^k)}.$$

Следовательно,

$$\|z^{k+1}\|_D^2 = \|z^k\|_D^2 - \frac{(Dz^k, B^{-1} Az^k)^2}{(DB^{-1} Az^k, B^{-1} Az^k)} \leq \|z^k\|_D^2.$$

Введем обозначения: $r^k = Az^k = Ax^k - f$ — невязка решения; $w^k = B^{-1} r^k = B^{-1} (Ax^k - f)$ — **поправка решения** на k -й итерации. Тогда

$$\tau_{k+1} = \frac{(w^k, Dz^k)}{(Dw^k, w^k)}; \quad x^{k+1} = x^k - \tau_{k+1} w^k.$$

Алгоритм решения таков: по заданному x^k находится невязка r^k , по ней — поправка $w^k = B^{-1}r^k$ и далее τ_{k+1} и x^{k+1} . Потом цикл повторяется.

Отметим, что на практике полноценное обращение матрицы B , как правило, не используют. Вместо этого очередную поправку w^k ищут как решение системы линейных уравнений $Bw^k = r^k$ каким-либо численным методом. В случае, например, прямых методов решения это позволяет сократить временные затраты на расчет w^k в 2 раза и более.

Словесное описание алгоритма решения вполне понятно, однако анализ полученных формул показывает, что реализовать его можно далеко не всегда. По заданному x^k без особого труда находится невязка r^k , а по ней — поправка w^k , для произвольного же оператора D , удовлетворяющего описанным условиям, напрямую вычислить Dz^k нельзя, так как погрешность z^k неизвестна. В результате невозможно найти параметр τ_{k+1} . Практически могут быть реализованы варианты метода только с такими операторами D , для которых можно вычислить Dz^k . Обычно их представляют в виде произведения операторов: $D = \dots \cdot A$. В этом случае $Dz^k = \dots \cdot r^k$ и новый итерационный параметр может быть вычислен.

3.2.3. Оценка скорости сходимости

Введем обозначение: $y^k = D^{1/2}z^k$, тогда $\|y^k\|^2 = \|z^k\|_D^2$ (D и $D^{1/2}$ — самосопряженные операторы) и погрешность решения СЛАУ на $(k+1)$ -м шаге

$$z^{k+1} = (E - \tau_{k+1}B^{-1}A)z^k.$$

Следовательно,

$$y^{k+1} = (E - \tau_{k+1}D^{1/2}B^{-1}AD^{-1/2})y^k = (E - \tau_{k+1}C)y^k,$$

где $C = D^{1/2}B^{-1}AD^{-1/2} = D^{-1/2}(DB^{-1}A)D^{-1/2}$. В результате

$$\|z^{k+1}\|_D^2 = \|(E - \tau_{k+1}C)y^k\|^2 = \|y^{k+1}\|^2.$$

Поскольку τ_{k+1} выбирают из условия минимума величины $\|z^{k+1}\|_D$, то

$$\begin{aligned} \|z^{k+1}\|_D &= \min_{\tau_{k+1}} \|(E - \tau_{k+1}C)y^k\| \leqslant \\ &\leqslant \min_{\tau_{k+1}} \|E - \tau_{k+1}C\| \|y^k\| = \min_{\tau_{k+1}} \|E - \tau_{k+1}C\| \|z^k\|_D. \end{aligned}$$

В итоге

$$\|z^m\|_D \leqslant \rho^m \|z^0\|_D, \quad \rho = \min_{\tau} \|E - \tau C\|.$$

Теорема 3.3. Пусть оператор $DB^{-1}A$ самосопряженный и выполнены условия

$$\gamma_1 D \leqslant DB^{-1}A \leqslant \gamma_2 D,$$

где $\gamma_1, \gamma_2 > 0$ — постоянные. Тогда

$$\rho = \frac{1 - \eta}{1 + \eta}, \quad \eta = \gamma_1/\gamma_2; \quad \|z^m\|_D \leqslant \left(\frac{1 - \eta}{1 + \eta}\right)^m \|z^0\|_D = \rho^m \|z^0\|_D.$$

◀ Поскольку

$$\gamma_1(Dx, x) \leqslant (DB^{-1}Ax, x) \leqslant \gamma_2(Dx, x), \quad x = D^{-1/2}y,$$

то получаем

$$\gamma_1(y, y) \leqslant (D^{-1/2}DB^{-1}AD^{-1/2}y, y) \leqslant \gamma_2(y, y),$$

откуда $\gamma_1 E \leqslant C \leqslant \gamma_2 E$.

Поскольку

$$\tau_{k+1} = \frac{(Cy^k, y^k)}{(Cy^k, Cy^k)},$$

то $\tau_{k+1} \geqslant 0$. Воспользуемся этим обстоятельством следующим образом. При $\tau \geqslant 0$ справедливо неравенство

$$-\gamma_2 \tau E \leqslant -\tau C \leqslant -\gamma_1 \tau E.$$

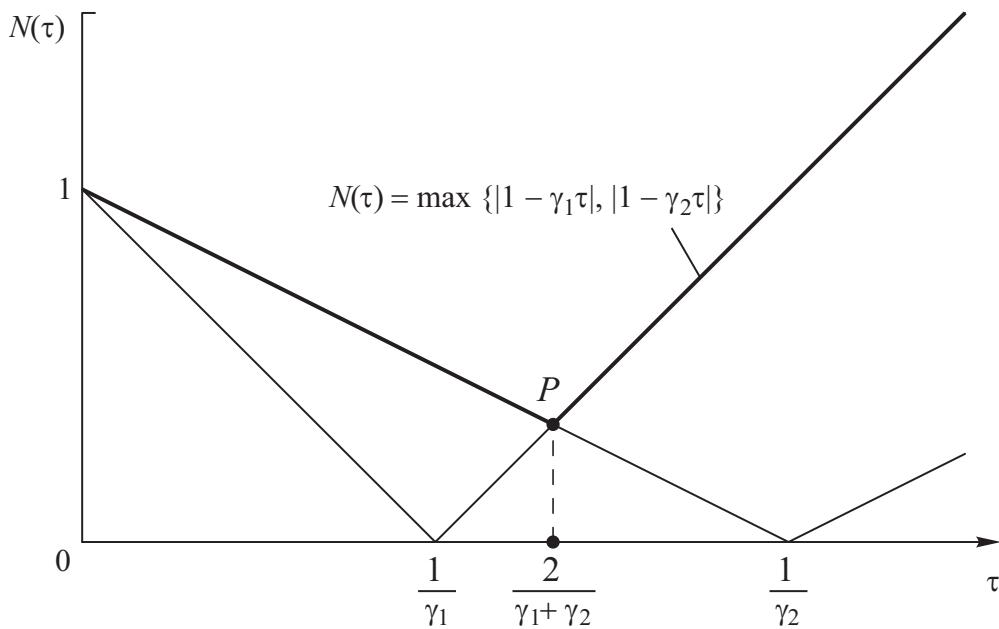


Рис. 3.5. Зависимость $N(\tau)$ для определения коэффициента перехода ρ

Отсюда получаем

$$(1 - \gamma_2\tau)E \leq E - \tau C \leq (1 - \gamma_1\tau)E.$$

Легко видеть, что в силу сделанных предположений оператор C — самосопряженный. Поэтому оценка оператора $E - \tau C$ позволяет получить оценку его спектра: спектр лежит на отрезке $[1 - \gamma_2\tau, 1 - \gamma_1\tau]$. Следовательно,

$$\|E - \tau C\| \leq \max \{|1 - \gamma_1\tau|, |1 - \gamma_2\tau|\} = N(\tau),$$

так как в случае самосопряженного оператора норма есть максимальное по модулю собственное значение.

Найдем значение параметра τ , минимизирующее $N(\tau)$ (искомая оценка есть минимум указанной величины по τ). В точке P (рис. 3.5) условие $1 - \gamma_1\tau = -1 + \gamma_2\tau$, т. е. $\tau = 2/(\gamma_1 + \gamma_2)$. Отсюда окончательно получаем, что

$$\rho \leq \min_{\tau} N = \frac{\gamma_2 - \gamma_1}{\gamma_2 + \gamma_1} = \frac{1 - \eta}{1 + \eta}. \quad \blacktriangleright$$

Следствие 3.4. Оптимальным значением параметра τ в обобщенном методе простых итераций

$$B \frac{x^{k+1} - x^k}{\tau} + Ax^k = f,$$

минимизирующим норму $\|z^k\|_D$ с операторами A, B, D , которые удовлетворяют условиям теоремы 3.3, является $\tau = 2/(\gamma_1 + \gamma_2)$.

3.2.4. Частные случаи методов вариационного типа

При выводе общих расчетных формул итерационных методов вариационного типа (см. 3.2.2) уже обсуждалась их реализуемость. Приведем примеры методов, для которых можно вычислить все необходимые величины.

1. Если $D = A^*A$, то норма погрешности $\|z^k\|_D^2 = (A^*Az^k, z^k) = (Az^k, Az^k) = (r^k, r^k)$. В результате получаем **метод минимальных невязок** (при условии, что A — невырожденный оператор). Параметры метода:

$$\tau_{k+1} = \frac{(w^k, A^*r^k)}{(Aw^k, Aw^k)} = \frac{(Aw^k, r^k)}{(Aw^k, Aw^k)}.$$

2. Если $D = A = A^* > 0$, то норма погрешности

$$\|z^k\|_D^2 = (Az^k, z^k)$$

вычисляется в энергетическом пространстве оператора A . В результате имеем **метод скорейшего спуска**. Параметры метода:

$$\tau_{k+1} = \frac{(w^k, r^k)}{(Aw^k, w^k)}.$$

3. Если $D = A^*B^{-1}A$, то получим **метод минимальных поправок**. В этом случае $Dz^k = A^*B^{-1}r^k = A^*w^k$, где B — самосопряженный положительно определенный оператор. Норма погрешности

$$\begin{aligned}\|z^k\|_D^2 &= (A^*B^{-1}Az^k, z^k) = (Az^k, B^{-1}Az^k) = \\ &= (BB^{-1}Az^k, B^{-1}Az^k) = (Bw^k, w^k).\end{aligned}$$

Параметры метода вычисляются по формуле

$$\tau_{k+1} = \frac{(w^k, A^*w^k)}{(B^{-1}Aw^k, Aw^k)}.$$

4. Если $D = B_0 = A^*B$, где $B_0 = B_0^*$, то получим **метод минимальных погрешностей**. В этом случае $B = (A^*)^{-1}B_0$. В результате имеем

$$\begin{aligned}\|z^k\|_D^2 &= \|z^k\|_{B_0}^2 = (A^*Bz^k, z^k); \\ Dz^k &= B_0z^k; \quad w^k = B^{-1}r^k = B_0^{-1}A^*r^k.\end{aligned}$$

Следовательно,

$$\begin{aligned}(w^k, Dz^k) &= (B_0^{-1}A^*r^k, B_0z^k) = (r^k, Az^k) = (r^k, r^k); \\ Dw^k &= A^*BB^{-1}Az^k = A^*r^k; \\ (Dw^k, w^k) &= (r^k, Aw^k),\end{aligned}$$

откуда

$$\tau_{k+1} = \frac{(r^k, r^k)}{(r^k, Aw^k)}.$$

Во всех четырех случаях предполагается, что операторы A , B (B_0) таковы, что D — самосопряженный положительно определенный оператор, $DB^{-1}A$ — самосопряженный оператор и выполнено неравенство $\gamma_1 D \leq DB^{-1}A \leq \gamma_2 D$ из теоремы 3.3. Тогда, чтобы уменьшить начальную погрешность (в норме D) в ε^{-1} раз, требуется выполнить $n_0(\varepsilon) = \ln(\varepsilon^{-1})/\ln(\rho^{-1})$ итераций.

Проблема заключается в том, что в реальных задачах значение ρ очень близко к единице (например, на практике отношение $\gamma_1/\gamma_2 \sim 10^{-4}$ считается достаточно большим).

Отметим, что для реализации итераций по рассмотренным методам вариационного типа, вообще говоря, нет необходимости знать (и хранить в оперативной памяти ЭВМ) полную матрицу A . Достаточно лишь иметь процедуру построения вектора Ax и иногда A^*x по вектору x . Это означает, что указанные методы можно применять даже тогда, когда матрица A доступна лишь посредством «черного ящика». Например, в тех случаях, когда матрица A не может быть вычислена в силу сложности решаемой задачи математического моделирования или когда она настолько велика, что ее невозможно целиком сохранить в оперативной памяти. В подобной ситуации, часто возникающей, например, при решении конструкторских задач, прямые методы неприменимы, а использование классических итерационных методов существенно затруднено.

3.3. Методы сопряженных направлений

Приведем расчетные формулы с их кратким пояснением для **двухшаговых (трехслойных) итерационных методов — методов сопряженных направлений**. При реализации этих методов необходимо знать приближенные решения, полученные на двух предыдущих итерациях. Рассматриваемые методы принадлежат к следующему классу итерационных методов:

$$B \frac{(x^{k+1} - x^k) + (1 - \alpha_{k+1})(x^k - x^{k-1})}{\tau_{k+1} \alpha_{k+1}} + Ax^k = f, \quad k = 1, 2, \dots$$

В отличие от одношаговых методов здесь имеется дополнительный итерационный параметр α_{k+1} . Кроме того, начать расчеты можно только при наличии двух начальных приближений x^0 и x^1 . Первое из них обычно берется произвольным, а второе вычисляется путем выполнения одной итерации того

же двухслойного итерационного метода вариационного типа, но при $k = 0$, $\alpha_1 = 1$. Если два предыдущих приближения известны и параметры найдены, то новое итерационное приближение находится без особого труда (оператор B^{-1} должен быть сравнительно легко обратим).

Итерационные параметры выберем так, чтобы при переходе от нулевого слоя к произвольному m -му операторная норма погрешности $\|z^m\|_D = (Dz^m, z^m)^{1/2}$ стала минимальной. Здесь D — некоторый *строго положительно определенный оператор*, задающий рассматриваемый метод вместе с матрицей B . Погрешность решения на k -й итерации $z^k = x^k - x$.

Постановка задачи о минимизации погрешности соответствует постановке задачи о построении *метода Ричардсона с чебышёвскими параметрами*. Из этого следует, что оценка нормы погрешности после m итераций в методе сопряженных направлений будет не менее точной, чем оценка погрешности метода Ричардсона.

Решение задачи минимизации дает следующие формулы для итерационных параметров:

$$\begin{aligned}\tau_{k+1} &= \frac{(Dw^k, z^k)}{(Dw^k, w^k)}, \quad k = 0, 1, \dots; \\ \alpha_1 &= 1; \quad \alpha_{k+1} = \left(1 - \frac{\tau_{k+1}}{\tau_k} \frac{(Dw^k, z^k)}{(Dw^k, w^k)} \frac{1}{\alpha_k}\right)^{-1}, \quad k = 1, 2, \dots.\end{aligned}$$

Здесь используется тот же набор обозначений, что и в случае одношаговых (двухслойных) итерационных методов вариационного типа (см. 3.2.2). Сходство двух- и трехслойных методов заключается не только в этом. Из расчетных формул следует, что выражения для итерационных параметров τ_{k+1} этих методов совпадают, а для вычисления α_{k+1} не требуется новых скалярных произведений. Таким образом, для нахождения итерационных

параметров в методах сопряженных направлений выполняется практически такое же количество действий.

Вычисление нового итерационного приближения несколько более трудоемко, чем в случае двухслойных методов. Однако это вполне компенсируется значительно большей скоростью сходимости. Как уже указывалось, она не меньше, чем у метода Ричардсона с чебышёвскими параметрами.

Более того, в конечномерном пространстве методы сопряженных направлений (при естественных ограничениях, накладываемых на применяемые операторы) сходятся за количество итераций, не превышающее размерности пространства.

Без излишней детализации приведем частные случаи методов сопряженных направлений. Каждый из них имеет свой одношаговый аналог.

1. Если $D = A^*A$, имеем **метод сопряженных невязок**.
2. Если $D = A = A^* > 0$, получаем **метод сопряженных градиентов**.
3. Если $D = A^*B^{-1}A$, имеем **метод сопряженных поправок**.
4. Если $D = B_0 = A^*B$, получаем **метод сопряженных погрешностей**.

Как и ранее, во всех случаях считается, что операторы A , B (B_0) таковы, что D — самосопряженный положительно определенный оператор, $DB^{-1}A$ — самосопряженный оператор и выполнено неравенство $\gamma_1 D \leq DB^{-1}A \leq \gamma_2 D$ из теоремы 3.3.

Отметим, что при решении плохо обусловленных задач вследствие погрешностей округления метод может и не сойтись за число итераций, равное порядку системы. В этом случае необходимо принимать специальные меры.

Кроме того, двухшаговые методы часто проявляют свойство «насыщения» погрешности: норма погрешности численного решения быстро убывает на первых итерациях, но затем выходит

на практически постоянное ненулевое значение («полочку»). В этом случае иногда помогают прерывание итерационного цикла и его повтор с полученного приближения. При этом снова выполняется один шаг по формулам одношагового метода, после чего повторяются итерации двухшагового.

3.4. Предобуславливание

Рассмотрим выбор матрицы B в канонической форме итерационных методов вариационного типа (см. 3.2.1). Удачный выбор такой матрицы может существенно ускорить итерационный процесс или даже обеспечить сходимость метода в том случае, когда матрица A настолько плохо обусловлена, что методы с $B = E$ расходятся.

Отметим, что при выборе $B = A$ итерационный процесс сходится за одну итерацию. Действительно, при выборе начального приближения $x^0 = 0$ и параметра $\tau_1 = 1$ каноническая форма метода

$$B \frac{x^1 - x^0}{\tau_1} + Ax^0 = f$$

сводится к решению системы

$$Bx^1 = f,$$

эквивалентной исходной. При этом, однако, задача нисколько не упрощается — матрицу B обратить столь же сложно, как и матрицу A . Поэтому матрицу B необходимо выбирать так, чтобы она, с одной стороны, была как можно «ближе» к матрице A , а с другой — чтобы матрица B^{-1} вычислялась как можно проще. Под «обращением матрицы» имеется в виду решение СЛАУ с этой матрицей.

Определение. Матрица B называется **матрицей предобуславливания** или **предобуславливателем** системы $Ax = f$,

если выполняются следующие условия:

- 1) матрица B самосопряженная и положительно определенная;
- 2) матрица $B^{-1}A$ хорошо обусловлена;
- 3) система $Bx = b$ решается легко.

Требования 2 и 3 в приведенном определении вступают между собой в противоречие — выбор $B = A$ дает хорошую обусловленность матрицы $B^{-1}A$, но не упрощает решение системы, а выбор $B = E$ делает решение системы $Bx = b$ тривиальным, но не изменяет числа обусловленности матрицы $B^{-1}A$.

Приведем несколько возможных вариантов выбора матрицы B .

1. Если диагональные элементы матрицы A сильно различаются по значению, то можно использовать простой **диагональный предобуславливатель** $B = \text{diag}(a_{11}, \dots, a_{nn})$, называемый еще **предобуславливателем Якоби**. Такой выбор дает число обусловленности произведения $B^{-1}A$, не более чем в n раз превышающее минимально возможное значение среди всех диагональных предобуславливателей.

2. Рассмотрим матрицу A как блочную матрицу с квадратными диагональными блоками A_{ii} :

$$A = \begin{pmatrix} A_{11} & \dots & A_{1k} \\ \dots & \dots & \dots \\ A_{k1} & \dots & A_{kk} \end{pmatrix}.$$

Тогда выбор матрицы

$$B = \begin{pmatrix} B_{11} & \dots & 0 \\ \dots & \dots & \dots \\ 0 & \dots & B_{kk} \end{pmatrix}$$

с блоками $B_{ii} = A_{ii}$ с точностью до множителя β дает минимальное число обусловленности матрицы $B^{-1/2}AB^{-1/2}$. Этот вариант называется **блочным предобуславливателем Якоби**.

3. Подобно *методу Якоби*, *метод релаксации* можно использовать в качестве (блочного) предобуславливателя.

4. Под *неполным LU-разложением матрицы A* понимают приближение $A \approx \tilde{L}\tilde{U}$, в котором матрицы \tilde{L} и \tilde{U} являются соответственно нижней и верхней треугольными и имеют тот же портрет (ненулевые элементы на тех же местах), что и матрица A . Использование произведения $\tilde{L}\tilde{U}$ дает высокоэффективный предобуславливатель, но требует дополнительных и довольно значительных затрат вычислительных ресурсов.

3.5. Итерационное уточнение решения

Численное решение СЛАУ $Ax = f$ *прямым* или *итерационным методом* является только приближением к точному. При этом чем «хуже» матрица системы (чем больше ее *число обусловленности*), тем дальше решение от точного. Полученная погрешность зависит также от разрядности ЭВМ, качества программной реализации алгоритма и пр. Однако в любом случае решение будет неточным.

Повысить точность приближенного решения можно способом *итерационного уточнения*. Выполним следующие этапы до сходимости (по некоторому критерию), положив для начала $x^0 = 0$. Итак, имеем цикл по k от нуля с указанным начальным приближением.

1. Находим невязку $r^k = f - Ax^k$.

2. Нормируем невязку, вычисляя $b^k = \frac{r^k}{\|r^k\|}$. В результате $\|b^k\| = 1$.

3. Решаем систему уравнений $Ay^k = b^k$ и определяем поправку y^k .

4. Находим новое приближение $x^{k+1} = x^k + y^k \|r^k\|$.

5. Проверяем, удовлетворяет ли полученное приближение критерию прекращения итераций.
6. Если x^{k+1} не удовлетворяет критерию прекращения итераций, то возвращаемся к этапу 1.

Отметим, что в результате нормирования на этапе 2 правой части системы $Ay^k = b^k$ (невязки) всегда решается одна и та же (по норме правой части) система на этапе 3, поэтому никаких дополнительных проблем малость правой части не привносит. Поскольку решается система с одной и той же матрицей, то обращение матрицы может быть выполнено один раз с использованием прямого метода, например *методом LU-разложения*, либо может быть выбран предобусловливатель, полученный на предыдущей итерации, если указанная система решается итерационным методом.

Наиболее ответственные этапы итерационного уточнения — расчет невязки и пересчет приближения. Здесь возможны как вычитание близких чисел, так и сложение чисел, порядки которых далеки друг от друга. Поэтому метод итерационного уточнения необходимо реализовывать по крайней мере в арифметике двойной точности.

Если число обусловленности решаемой системы уравнений не слишком велико, то описанный итерационный метод сходится сравнительно быстро. При этом даже при относительно большом числе обусловленности системы процесс итерационного уточнения оказывается сходящимся. Под сходимостью чаще всего понимается установление значащих цифр в приближенном решении.

Отметим, что алгоритм итерационного уточнения решения является буквальным применением метода Ньютона для решения нелинейных систем уравнений к решению СЛАУ в приближенной арифметике.

3.6. Решение проблемы собственных значений

Определим собственные значения матрицы A :

$$Ax_k = \lambda_k x_k.$$

Определение. *Полной проблемой собственных значений* называется задача отыскания всех собственных значений и собственных векторов матрицы A , *частичной (ограниченной)* — задача отыскания лишь их части.

Ограничимся задачей нахождения минимального и максимального собственных значений *строго положительно определенной матрицы* A , имеющей ортонормированный базис из собственных векторов.

Пусть λ_i — собственные значения оператора A , причем

$$0 < \alpha(A) = \lambda_1 < \lambda_2 \leq \dots \leq \lambda_{n-1} < \lambda_n = \beta(A),$$

а e_i , $i = \overline{1, n}$, — собственные векторы.

Утверждение 3.1. Построим следующий итерационный процесс:

$$\varphi_0 = g; \quad \varphi_{k+1} = \frac{A\varphi_k}{\|\varphi_k\|}.$$

Тогда $\beta(A) = \lim_{k \rightarrow \infty} \|\varphi_k\|$.

◀ Разложим φ_0 по собственным векторам матрицы A :

$$\varphi_0 = g = \sum_{i=1}^n c_i e_i.$$

Тогда

$$A^k \varphi_0 = \sum_{i=1}^n c_i \lambda_i^k e_i; \quad \|A^k \varphi_0\|^2 = \sum_{i=1}^n c_i^2 \lambda_i^{2k}.$$

Следовательно,

$$\varphi_{k+1} = \frac{A\varphi_k}{\|\varphi_k\|} = \frac{A^2 \varphi_{k-1}}{\|\varphi_{k-1}\|} \frac{\|\varphi_{k-1}\|}{\|A\varphi_{k-1}\|} = \frac{A^3 \varphi_{k-2}}{\|\varphi_{k-2}\|} \frac{\|\varphi_{k-2}\|}{\|A^2 \varphi_{k-2}\|} = \dots = \frac{A^{k+1} \varphi_0}{\|A^k \varphi_0\|},$$

причем

$$\frac{\|A^{k+1}\varphi_0\|^2}{\|A^k\varphi_0\|^2} = \lambda_n^2 + O\left(\frac{\lambda_{n-1}^{2k}}{\lambda_n^{2k-2}}\right),$$

откуда

$$\frac{\|A^{k+1}\varphi_0\|}{\|A^k\varphi_0\|} = \beta(A) + O\left(\left(\frac{\lambda_{n-1}}{\lambda_n}\right)^{2k}\right).$$

Значит,

$$\beta(A) = \lim_{n \rightarrow \infty} \|\varphi_n\|. \quad \blacktriangleright$$

Следствие 3.5. Для любой строго положительно определенной матрицы A имеет место равенство

$$\alpha(A) = \beta(A) - \beta(\beta(A)E - A).$$

◀ Рассмотрим оператор $B = \beta(A)E - A$. Его максимальное собственное значение $\beta(B) = \beta(A) - \alpha(A)$. Отсюда $\alpha(A) = \beta(A) - \beta(B)$. Для нахождения $\beta(B)$ используется тот же алгоритм, что и в утверждении 3.1. ►

Построенный в утверждении 3.1 итерационный процесс называется **степенным методом** нахождения наибольшего по модулю собственного значения матрицы A . Он может быть применен и к произвольной вещественной матрице A . Для определения других собственных значений и собственных векторов степенной метод может быть применен к матрице $(A - \sigma E)^{-1}$ с использованием сдвига σ . Наибольшим по модулю собственным значением матрицы $(A - \sigma E)^{-1}$ является число $1/(\lambda_i - \sigma)$, где λ_i — собственное значение матрицы A , ближайшее к σ . Этот алгоритм называется **обратной итерацией**.

Замечание 3.3. Для поиска $(n - 1)$ -го (следующего после максимального) собственного значения необходимо исключить из получаемых в процессе итераций векторов часть, соответствующую n -му собственному вектору. Это позволит выполнить

ту же процедуру, но уже в подпространстве, ортогональном этому вектору. В частности, можно использовать следующие соотношения:

$$A^{k+1}\varphi_0 - \lambda_n A^k \varphi_0 = \sum_{i=1}^{n-1} c_i (\lambda_i^{k+1} - \lambda_i^k \lambda_n) e_i;$$

$$A^k \varphi_0 - \lambda_n A^{k-1} \varphi_0 = \sum_{i=1}^{n-1} c_i (\lambda_i^k - \lambda_i^{k-1} \lambda_n) e_i.$$

Если λ_{n-1} отделено от λ_{n-2} , то отношение норм двух последних векторов должно стремиться к λ_{n-1} при $k \rightarrow \infty$.

Замечание 3.4. Теоремы о кругах Гершгорина позволяют тривиально оценить границы собственных значений. Довольно часто верхняя граница может быть оценена достаточно надежно. Нижняя же граница, как правило, оценивается неточно.

3.7. Регуляризация плохо обусловленных систем линейных алгебраических уравнений

Рассмотрим СЛАУ

$$Ax = f.$$

Пусть *число обусловленности матрицы A* велико. В этом случае формально полученное решение рассматриваемой задачи несет в себе столько погрешностей вычислений, что теряет всякий смысл. Необходима иная постановка исходной задачи.

Следствием точной постановки является равенство

$$\|Ax - f\|^2 = 0.$$

Однако в случае плохо обусловленной системы вполне разумно считать, что

$$\|Ax - f\|^2 \approx 0.$$

Для определенности, т. е. выделения единственного решения, требуется добавить условие, которого не было в исходной постановке задачи. Такое условие может быть взято как минимум отклонения от заданного вектора x_0 и минимум нормы разности $Ax - f$ в виде

$$\Phi(x, x_0) = \|Ax - f\|^2 + \alpha\|x - x_0\|^2 \rightarrow \min.$$

Здесь $\alpha > 0$ — управляющий параметр, называемый **параметром регуляризации**. Рассматриваемый метод называется **методом регуляризации СЛАУ**.

Преобразуем последнее выражение:

$$\begin{aligned} \Phi(x, x_0) = & (x, A^*Ax) - 2(Ax, f) + (f, f) + \\ & + \alpha(x, x) - 2\alpha(x, x_0) + \alpha(x_0, x_0) \rightarrow \min. \end{aligned}$$

Нахождение экстремума этого функционала (варьирование x) дает следующее уравнение для x :

$$A^*Ax - A^*f + \alpha Ex - \alpha Ex_0 = 0,$$

или

$$(A^*A + \alpha E)x = A^*f + \alpha Ex_0.$$

Здесь матрица A^*A положительно определена. Если $\alpha > 0$, то при его достаточно большом значении найти решение задачи, например, методом Гаусса не составит труда. Однако $x = x(x_0, \alpha)$. Как выбрать x_0 и α ?

Вектор x_0 выбирают из соображений физического или технического характера. Если таких соображений нет, то обычно полагают, что $x_0 = 0$.

Рассмотрим выбор α . Если α взять слишком малым, то задача получится плохо обусловленной, если слишком большим, то норма невязки $\|Ax - f\|$ окажется большой, т. е. решение исходной СЛАУ будет неточным. Обычно α выбирают по принципу

невязки: ищут такое α , чтобы выполнялось условие

$$\|Ax - f\| \approx \|\delta f\| + \|\delta Ax\|,$$

где δf и δA — априорно заданные погрешности правой части СЛАУ и оператора соответственно.

Здесь приведен один из простейших примеров *метода регуляризации* некорректно поставленных задач **A.Н. Тихонова**.

3.8. Хранение больших разреженных матриц

При дискретизации задач математической физики возникают СЛАУ с большим числом неизвестных n . В современных многомерных задачах значение n может достигать многих тысяч и миллионов. Матрица A возникшей СЛАУ формально имеет n^2 элементов. Даже для современных ЭВМ объем этой информации может оказаться недопустимо большим. Однако чаще всего *матрица* бывает редко заполненной (*разреженной*), т. е. содержит большое количество нулей. Общее количество N ненулевых элементов в таких задачах удовлетворяет условию $N \ll n^2$. Ясно, что хранить большое количество нулей бессмысленно.

Опишем различные способы хранения таких матриц.

1. Для хранения матрицы A в виде двумерного массива элементов $\{a_{ij}\}$, $i, j = \overline{1, n}$, требуется «помнить» n^2 элементов, большинство из которых нули. Вариант неприемлем.

2. Простейший вариант — хранение только ненулевых элементов матрицы A в виде линейного массива $\{a_k\}$, $k = \overline{1, N}$, состоящего чаще всего из действительных чисел. Проще всего ненулевые элементы упорядочить по строкам слева направо. При этом для каждого элемента необходимо указать номера строки и столбца, которые он занимает в исходной матрице: $\{row_a_k\}$, $k = \overline{1, N}$, и $\{col_a_k\}$, $k = \overline{1, N}$. Данные массивы целочисленные. В этом варианте требуется хранить $3N$ значений.

3. Вариант, требующий минимальных ресурсов памяти, — хранение только ненулевых элементов матрицы A в виде линейного массива $\{a_k\}$, $k = \overline{1, N}$, и одного целочисленного массива $\{nr_a_k\}$, $k = \overline{1, N}$; последний массив содержит номер соответствующего элемента в исходной двумерной матрице при ее хранении, например, в виде одномерного массива по столбцам. В этом случае номер k -го элемента $nr_a_k = i_k + (j_k - 1)n$. Здесь i_k , j_k — номера строки и столбца рассматриваемого элемента. При таком варианте требуется хранить $2N$ значений. Однако очевидно, что при работе с матрицей, хранящейся в таком виде, придется постоянно пересчитывать номера строки и столбца каждого элемента.

4. Оптimalен по требуемым ресурсам памяти и удобству работы так называемый разреженный строчный формат хранения данных. При этом необходимо хранить только упорядоченные по строкам слева направо ненулевые элементы матрицы A в виде линейного массива $\{a_k\}$, $k = \overline{1, N}$, массив $\{cln_a_k\}$, $k = \overline{1, N}$, с номерами столбцов элементов в исходной матрице и массив $\{rf_a_k\}$, $k = \overline{1, n+1}$. Два последних массива — целочисленные. В массиве $\{rf_a_k\}$ хранятся номера первого элемента k -й строки при сплошной нумерации линейного одномерного массива. Таким образом, данный массив указывает, что элементы с номерами от rf_a_k до $rf_a_{k+1} - 1$ (включительно) при сплошной нумерации одномерного массива соответствуют k -й строке исходной матрицы. Отсюда следует, что необходимо положить $rf_a_1 = 1$, $rf_a_{n+1} = N + 1$. В этом варианте требуется хранить $2N + n + 1$ значений.

Вопросы и задания

1. Какой метод решения СЛАУ называется итерационным?
Что называется погрешностью решения? Что такое невязка?
Какой итерационный процесс называется сходящимся?

2. Что означает найти решение СЛАУ с заданной точностью? Что такое критерий прекращения итераций? Приведите примеры критериев прекращения итераций.
3. Запишите каноническую форму двухслойных (одношаговых) итерационных методов решения СЛАУ. Какой метод называется явным, а какой неявным? Приведите примеры стационарных и нестационарных методов.
4. Сформулируйте условия сходимости стационарных двухслойных (одношаговых) методов. Как влияет выбор начального приближения на сходимость итерационного процесса?
5. Сформулируйте условия сходимости методов простой итерации, Зейделя и релаксации.
6. Какие методы решения СЛАУ называют методами вариационного типа? В чем заключается идея этих методов?
7. Каковы расчетные формулы итерационных методов решения СЛАУ вариационного типа? Приведите примеры таких методов.
8. Сформулируйте условия сходимости итерационных методов решения СЛАУ вариационного типа.
9. Что называется предобуславливателем? Приведите примеры предобуславливателей матриц.
10. Что такое полная и ограниченная проблемы собственных значений?
11. Сформулируйте алгоритм поиска максимального собственного значения симметричной положительно определенной матрицы. Как можно его применить для поиска остальных собственных значений этой матрицы?

Библиографические комментарии

Материал, приведенный в данной главе, применяется при численном решении практически всех задач математической физики. Почти в каждой из них требуется решить систему линейных уравнений.

В книге [69] подробно описаны различные итерационные методы решения СЛАУ. Уделено внимание решению систем уравнений с несамосопряженным оператором, рассмотрены случаи незнакоопределенного и вырожденного операторов. Представлены специализированные алгоритмы для решения некоторых классов задач с большим количеством действий и характеристик.

В настоящем пособии проблема собственных значений матриц и методы ее решения изложены лишь в общих чертах, очень кратко. Однако ряд задач науки и техники практически и состоит в решении данной проблемы в конкретном случае. Укажем специализированные по данной теме книги: это [44] и [78]. Много полезной информации можно найти в работах [9–11, 28, 35, 37].

Теория решения некорректно поставленных задач уже превратилась в самостоятельный раздел прикладной математики и математической физики. Представленное в данной главе описание метода регуляризации может быть существенно расширено с помощью работ [57, 67, 74, 75].

Алгоритмы вычислительной линейной алгебры постоянно совершенствуются: создаются новые и модернизируются старые, классические. Например, метод Гаусса применяют для решения больших разреженных СЛАУ со специальными алгоритмами сжатия полосы, содержащей ненулевые элементы [30].

В качестве литературы, посвященной технологии работы с матрицами специального вида, укажем такие работы, как, например, [31, 33, 34, 63, 77, 83, 85].

Современные алгоритмы вычислительной линейной алгебры представлены также в книге [24]. Развитие ЭВМ с параллельной архитектурой привело к появлению алгоритмов, предназначенных для использования именно на таких машинах [18, 31, 60].

4. МЕТОДЫ ИНТЕРПОЛИРОВАНИЯ ФУНКЦИЙ

Представлены варианты одномерной и многомерной интерполяции функций. Рассмотрена глобальная и локальная полиномиальная интерполяция. Описаны интерполяционные полиномы в форме Лагранжа и Ньютона, полином Эрмита. Рассмотрена задача о наилучшем приближении. Исследованы сходимость интерполяции при увеличении числа точек, устойчивость интерполяции по отношению к погрешностям исходной функции, зависимость погрешности интерполяции от гладкости функции. Введено понятие насыщаемости алгоритма, в данном случае — интерполяции. Описан алгоритм сплайн-интерполяции. Изложены простейшие способы двумерной интерполяции, в том числе с помощью конечных элементов.

4.1. Постановка задачи и простейшие методы интерполирования функций

4.1.1. Основные определения

Задача реконструкции (или аппроксимации) функции по ограниченному набору данных о ней не менее важна, чем задача решения СЛАУ. Во многих прикладных задачах (например, при решении дифференциальных уравнений) искомый результат в полном виде может быть представлен не одним числом, а функцией $y = f(x)$, где x — одна или несколько независимых переменных из некоторого множества Ω . Однако при численном решении задачи такое представление решения зачастую невозможно и результат выдается в виде набора значений функции для некоторого набора значений аргумента. Иными словами, часто доступная информация о функции — это значения функции $f(x)$,

заданные на некоторой сетке Ω_h (дискретное множество значений аргумента x из Ω). Тогда в случае, если необходимо найти значение решения в других точках Ω , требуется задать некоторое правило, которое позволит по значениям функции в узлах сетки найти приближенное значение функции в точках $\Omega \setminus \Omega_h$, не вошедших в нее. Дополнительные сложности возникают в том случае, если значения $f(x)$ известны неточно.

Определение. Конечное множество дискретных значений аргумента x из Ω со связями между ними называется **сеткой**, которую в дальнейшем будем обозначать $\Omega_h = \{x_i, i = \overline{0, N}\}$. При этом x_i называется **элементом** или **узлом сетки**.

Определение. **Функция**, определенная только в узлах сетки, называется **сеточной**.

Определение. Процедура, которая ставит в соответствие функции $f(x)$ ее сеточные значения f_i в точках x_i , называется **ограничением функции на сетку** (часто обозначается R , от англ. restriction — ограничение), т. е.

$$R: f(x) \rightarrow \{f_i\}.$$

Как правило, R выбирают так, что $f_i = f(x_i)$, однако способов ограничения функции на сетку бесконечно много. В результате ограничения создают таблицы, графики, схемы. Такой способ представления результата весьма распространен. Часто возникает обратная задача: по ограниченной информации о решении, выраженным функцией $f(x)$, вычислить значения этой функции в точках, не вошедших в сетку Ω_h . Тогда по заданным значениям $\{f_i\}$ требуется построить некоторую функцию $\tilde{f}(x)$, которая являлась бы приближением исходной функции $f(x)$.

Определение. Процедура построения \tilde{f} по $\{f_i\}$ называется **интерполяцией** (реконструкцией, аппроксимацией). Соответствующий оператор обозначим I (от англ. interpolation). Функция \tilde{f} называется **интерполянтом**.

В итоге имеем следующую схему:

$$f(x) \xrightarrow{R} \{f_i\} \xrightarrow{I} \tilde{f}(x).$$

При этом необходимо найти норму погрешности $\|\tilde{f} - f\|$. Ясно, что наиболее сложно ее определить в случае неизвестной функции $f(x)$.

Определение. Термин «интерполяция» в узком смысле обычно употребляют, если значение аргумента восстанавливаемой функции находится внутри области, ограниченной точками сетки, на которой задана функция. Если же значение аргумента выходит за границы этой области, то процедура построения \tilde{f} по $\{f_i\}$ называется **экстраполяцией**.

Вариантов построения $\tilde{f}(x)$ по заданной сетке Ω_h и сеточной функции $\{f_i\}$ бесконечно много даже в простейшем, одномерном, случае. Изучение задачи интерполяирования начнем именно с одномерного случая.

Рассмотрим отрезок $\Omega = \{x: a \leq x \leq b\}$, на котором задана сетка

$$\Omega_h = \{a = x_0 < x_1 < x_2 < \dots < x_n = b\}.$$

Пусть в точках сетки Ω_h известны значения функции f_i . Необходимо построить по ним функцию $\tilde{f}(x)$ так, чтобы $\tilde{f}(x_i) = f_i$. Возможны разные способы решения этой задачи, например с помощью кусочно-линейной, полиномиальной, тригонометрической или сплайн-интерполяции.

4.1.2. Кусочно-линейная интерполяция

Потребуем, чтобы помимо выполнения в узлах сетки равенства $\tilde{f}(x_i) = f_i$ функция $\tilde{f}(x)$ была линейна на каждом участке $[x_{i-1}, x_i]$, $i = \overline{1, n}$.

Очевидно, что этим условиям удовлетворяет кусочно-линейная функция

$$\tilde{f}(x) = \frac{1}{x_i - x_{i-1}} \left[(x - x_{i-1}) f_i - (x - x_i) f_{i-1} \right], \quad x \in [x_{i-1}, x_i].$$

Введем следующие функции:

$$\varphi_0(x) = \begin{cases} \frac{x_1 - x}{x_1 - x_0}, & x \in [x_0, x_1], \\ 0, & x \geq x_1; \end{cases}$$

$$\varphi_i(x) = \begin{cases} \frac{x - x_{i-1}}{x_i - x_{i-1}}, & x \in [x_{i-1}, x_i], \\ \frac{x_{i+1} - x}{x_{i+1} - x_i}, & x \in [x_i, x_{i+1}], \quad i = \overline{1, n-1}; \\ 0, & x \notin [x_{i-1}, x_{i+1}], \end{cases}$$

$$\varphi_n(x) = \begin{cases} 0, & x \leq x_{n-1}, \\ \frac{x - x_{n-1}}{x_n - x_{n-1}}, & x \in [x_{n-1}, x_n]. \end{cases}$$

Функции $\varphi_i(x)$ являются простейшими одномерными кусочно-линейными базисными функциями конечных элементов (рис. 4.1).

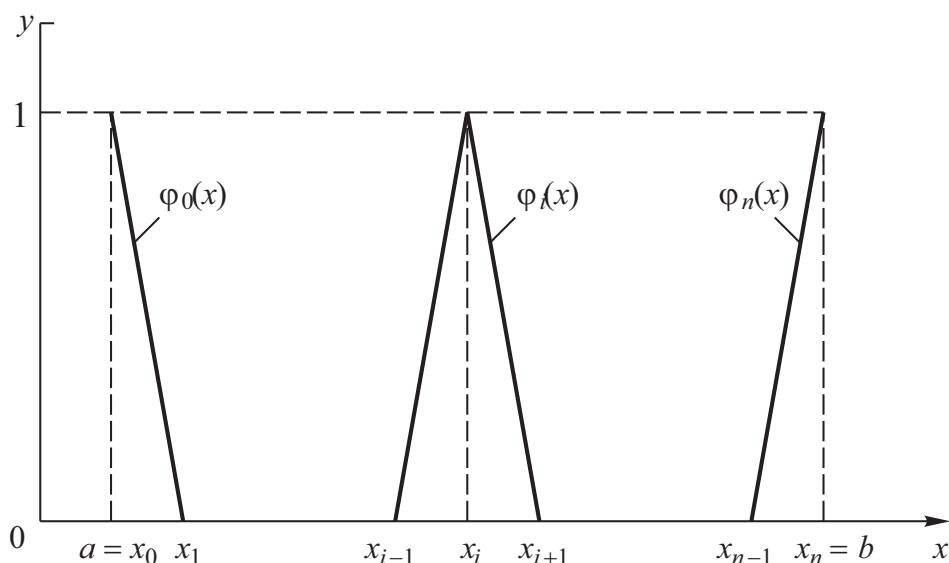


Рис. 4.1. Одномерные кусочно-линейные базисные функции конечных элементов

Определение. Отрезок $[x_{i-1}, x_i]$ с парой функций $\varphi^{(1)} = \frac{x - x_{i-1}}{x_i - x_{i-1}}$ и $\varphi^{(2)} = \frac{x - x_i}{x_{i-1} - x_i}$ называется **одномерным линейным конечным элементом**, а функции $\varphi^{(1)}$ и $\varphi^{(2)}$ — **функциями формы** данного конечного элемента. Заданная выше функция $\varphi_i(x)$ называется **базисной функцией** i -го узла конечно-элементной сетки Ω_h .

В более сложных случаях конечный элемент может содержать более двух узлов и, соответственно, функций формы, а базисная функция может быть задана как на двух конечных элементах, так и на одном. Аппарат конечно-элементной реконструкции функций с базисными функциями произвольного порядка хорошо разработан и подробно описан в литературе. Он лежит в основе большого числа современных программных пакетов инженерного анализа.

Используя построенные базисные функции, запишем решение задачи кусочно-линейной интерполяции в виде

$$\tilde{f}(x) = \sum_{i=0}^n \varphi_i(x) f_i.$$

Теорема 4.1. Пусть $f(x)$ — липшиц-непрерывная функция с постоянной q на отрезке $[a, b]$, т. е.

$$\forall x', x'' \in [a, b]: |f(x') - f(x'')| \leq q|x' - x''|.$$

Тогда справедлива оценка погрешности кусочно-линейной интерполяции

$$|f(x) - \tilde{f}(x)| \leq qh/2, \quad x \in [a, b],$$

где $h = \max_{1 \leq i \leq n} |x_i - x_{i-1}|$. Эта оценка неулучшаема на данном классе функций.

◀ Пусть $x \in [x_{i-1}, x_i]$, $h_i = x_i - x_{i-1}$, $x = x_{i-1} + \alpha h_i$, $\alpha = \frac{x - x_{i-1}}{h_i}$,

где $0 \leq \alpha \leq 1$. Тогда $\tilde{f}(x) = \alpha f_i + (1 - \alpha)f_{i-1}$ и

$$\begin{aligned} |f(x) - \tilde{f}(x)| &= |\alpha f_i + (1 - \alpha)f_{i-1} - \alpha f_i - (1 - \alpha)f_{i-1}| \leq \\ &\leq \alpha|f_i - \tilde{f}_i| + (1 - \alpha)|f_{i-1} - \tilde{f}_{i-1}| \leq \\ &\leq \alpha q|x - x_i| + (1 - \alpha)q|x - x_{i-1}| = \\ &= \alpha q(1 - \alpha)h_i + (1 - \alpha)q\alpha h_i = 2q\alpha(1 - \alpha)h_i \leq \frac{1}{2}qh_i. \end{aligned}$$

Отсюда

$$\|f - \tilde{f}\|_C \leq qh/2, \quad \forall x \in [a, b].$$

Данная оценка неулучшаема на функциях рассматриваемого класса. Покажем это на примере функции, для которой реализуется строгое равенство. Пусть $n = 1$, $a = -1$, $b = 1$, $f(x) = |x|$ — липшиц-непрерывная с постоянной $q = 1$ функция: для $x', x'' \geq 0$ (или $x', x'' \leq 0$) это очевидно ($|f'(x)| = 1$). При $x' \geq 0$, $x'' \leq 0$ выполнены следующие соотношения:

$$\begin{aligned} |f(x') - f(x'')| &= |x' - |x''|| = |x' + x''| \leq \\ &\leq |x'| + |x''| = x' - x'' = |x' - x''|. \end{aligned}$$

Обратная ситуация рассматривается аналогично.

Для данного случая $h = 2$, $\tilde{f} = 1$, поэтому

$$\max_{[-1;1]} |f - \tilde{f}| = 1 = qh/2$$

и достигается в точке $x = 0$. ►

4.1.3. Многовариантность интерполяции

Ранее был рассмотрен один из наиболее простых и часто применяемых приемов реконструкции функции. Однако вариантов интерполяции и способов восстановления функции $\tilde{f}(x)$ бесконечно много.

Пример 4.1. Пусть $x_0 = -1$, $x_1 = 0$, $x_2 = 1$ и во всех узлах $f_i = 0$. Тогда функция

$$\tilde{f}(x) = c(x+1)^\alpha x^\beta (x-1)^\gamma$$

для произвольных (с очевидными оговорками) положительных α , β , γ и любого значения c проходит через три точки (x_i, f_i) , $i = 0; 1; 2$. Если $f_i \neq 0$, то к любой функции, проходящей через эти три точки, можно добавить еще \tilde{f} . Полученная кривая также будет проходить через точки (x_i, f_i) , $i = 0; 1; 2$. •

Существует целый набор возможностей и методик, позволяющих решить задачу реконструкции функции с учетом требований исследователя и решаемой задачи. Например, часто результирующая функция становится более гладкой, если выполнить какую-нибудь замену переменных:

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

Ее необходимо либо угадывать, либо выбирать исходя из внешних (физических, технических) соображений. Типичный пример подобных замен — использование логарифмической шкалы при построении графиков.

Возможны интерполяции с помощью рациональных функций (например, дробно-линейных) или функциями любого параметрического семейства, соответствующего смыслу задачи. В этом случае интерполяント представляют в виде $F(a_0, \dots, a_n, x)$, а параметры a_0, \dots, a_n находят из условия $F(a_0, \dots, a_n, x_i) = f(x_i)$, $i = \overline{0, n}$, или любого другого условия реконструкции исходной функции.

Пример 4.2. Выполним интерполяцию функции, заданной в табличном виде:

x_i	2	6	11
$f(x_i)$	20	12	11

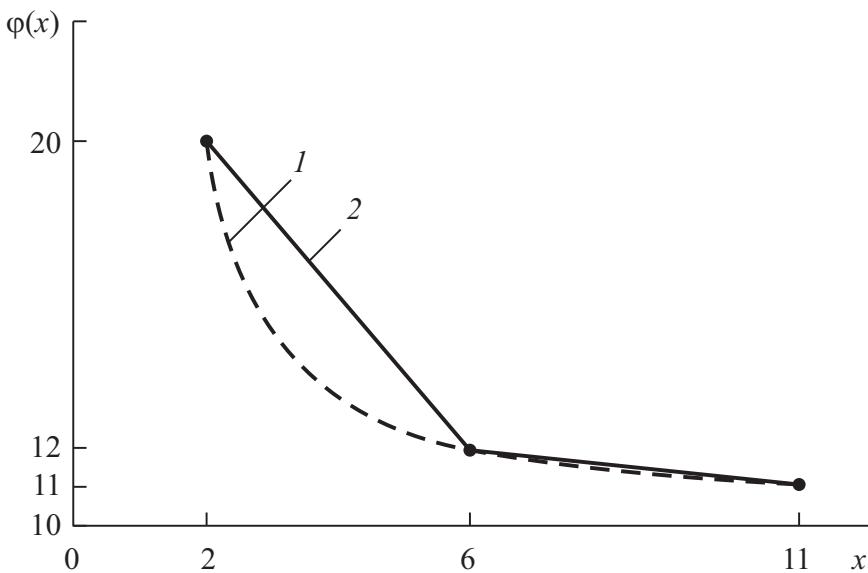


Рис. 4.2. Реконструкции функции из примера 4.2:
1 —дробно-линейная; 2 —кусочно-линейная

Возьмем интерполирующую функцию (интерполянт) в виде

$$\varphi(x) = \frac{x+a}{bx+c},$$

где a, b, c — неопределенные коэффициенты, и получим условия прохождения интерполянта через заданные точки:

$$\frac{2+a}{2b+c} = 20; \quad \frac{6+a}{6b+c} = 12; \quad \frac{11+a}{11b+c} = 11.$$

Решив эту систему, получим $a = 0$, $b = 1/10$, $c = -1/10$ и интерполянт

$$\varphi(x) = \frac{10x}{x-1}.$$

Для сравнения на рис. 4.2 изображена функция $\varphi(x)$ (кривая 1) и кусочно-линейный интерполянт (кривая 2). •

Интерполяция может выполняться как по интервалам, так и на всем отрезке, можно требовать точного прохождения приближающей (аппроксимирующей) функции через заданные точки, а можно и не требовать (метод наименьших квадратов

приближения функции, число параметров которого меньше числа точек сетки $n + 1$).

Замечание 4.1. Пусть задана сеточная функция $\{f_i\}$, $i = \overline{0, n}$, и система линейно независимых функций $\{\varphi_j(x)\}$, $j = \overline{0, m}$, линейная комбинация которых дает интерполянт, т. е. интерполирующую функцию. Если $n > m$, то задача интерполяции может стать переопределенной. В этом случае можно рассмотреть задачу о наилучшем приближении. Суть ее заключается в поиске такого обобщенного полинома

$$\varphi(x) = \sum_{j=0}^m a_j \varphi_j(x_i),$$

норма отклонения которого в узлах сетки $\{x_i\}$ от точного значения функции $\{f_i\}$ минимальна. Другими словами, необходимо найти такие коэффициенты $\{a_j\}$, чтобы норма вектора погрешности $r = (r_0, r_1, \dots, r_n)^T$, где $r_i = \varphi(x_i) - f_i$, была минимальна. В зависимости от выбора нормы получим разные задачи: при норме $\|r\| = \left(\sum_{i=0}^n r_i^2 \right)^{1/2}$ задачу о наилучшем среднеквадратичном приближении, а при норме $\|r\| = \max_i |r_i|$ — о наилучшем равномерном приближении.

Если $n = m$, то решение задачи о наилучшем приближении совпадает с решением задачи интерполяции, так как в этом случае требование $\|r\| = 0$ приводит к условию $\varphi(x_i) = f_i$, $i = \overline{0, n}$.

Пример 4.3. Рассмотрим линейную функцию, которая наилучшим образом приближает (в среднеквадратичном смысле) функцию $f(x)$, заданную своими тремя значениями f_0, f_1, f_2 в точках x_0, x_1, x_2 . Для простоты расстояние между точками сетки примем одинаковым, т. е. $x_2 - x_1 = x_1 - x_0 = h$. Будем искать приближающую (аппроксимирующую) функцию в виде

$\varphi(x) = a + b(x - x_1)$, неизвестные коэффициенты a, b при этом находим из условия

$$\delta^2 = \sum_{k=0}^2 (\varphi(x_k) - f_k)^2 \rightarrow \min.$$

В результате получим решение задачи в виде

$$a = \frac{1}{3}(f_0 + f_1 + f_2); \quad b = \frac{1}{2h}(f_2 - f_0),$$

при этом минимальное среднеквадратичное отклонение

$$\delta^2 = \frac{1}{6}h^4(f_{\bar{x}x})^2,$$

где $f_{\bar{x}x} = \frac{1}{h^2}(f_0 - 2f_1 + f_2)$.

Величина $f_{\bar{x}x}$, называемая разностной производной второго порядка, при условии определенной гладкости функции аппроксимирует обычную вторую производную и равна ее значению в некоторой средней точке на отрезке $[x_0, x_2]$. •

Определение. Алгоритм построения аппроксимирующей функции из условия минимума среднеквадратичного отклонения называется **методом наименьших квадратов**.

Замечание 4.2. Задача интерполяции является вариантом общей задачи приближения заданной функции $f(x)$ на каком-то участке суммой вида $\sum_{i=0}^n c_i \varphi_i(x)$. Как известно, в случае тригонометрических полиномов в результате минимизации нормы погрешности в пространстве L_2 получается частичная сумма ряда Фурье.

Наибольшее распространение на практике получили полиномиальная интерполяция, тригонометрическая и сплайн-интерполяция.

4.2. Полиномиальная интерполяция

4.2.1. Обоснование полиномиальной интерполяции

В основе полиномиального приближения функции $f(x)$ лежит следующая теорема.

Теорема 4.2 (Вейерштрасса). Для любой непрерывной на отрезке $[a, b]$ функции $f(x)$ существует полином $P_n(x)$, приближающий $f(x)$ с любой наперед заданной точностью:

$$\forall \varepsilon > 0, f(x) \in C[a, b] \quad \exists P_n(x) : \|f - P_n\|_C < \varepsilon.$$

Построим интерполяント следующим образом. Будем искать полином степени n вида

$$P_n(x) = a_0 + a_1x + \dots + a_nx^n,$$

проходящий через $n + 1$ точек (x_i, f_i) , $i = \overline{0, n}$. Для определения параметров a_k , $k = \overline{0, n}$, получим систему из $n + 1$ линейных уравнений с $n + 1$ неизвестными:

$$a_0 + \sum_{k=1}^n a_k x_i^k = f_i, \quad i = \overline{0, n}.$$

Определение. **Полином** $P_n(x)$ такой, что $P_n(x_i) = f_i$, $i = \overline{0, n}$, называется **интерполяционным**, а процедура его построения — **интерполяцией Лагранжа**.

Теорема 4.3. Интерполяционный полином $P_n(x)$ существует и является единственным (при $x_i \neq x_j$, $i \neq j$).

◀ Для построения интерполяционного полинома необходимо решить систему линейных уравнений $a_0 + \sum_{k=1}^n a_k x_i^k = f_i$, $i = \overline{0, n}$, и найти коэффициенты a_k . Определитель матрицы данной системы есть **определитель Вандермонда**, который отличен от нуля, если

точки сетки не совпадают. Отсюда следует, что решение задачи определения коэффициентов полинома существует и единственно.

Приведем иное доказательство единственности. Допустим, что утверждение теоремы неверно. Тогда существуют хотя бы два разных полинома $P_n^{(1)}(x)$ и $P_n^{(2)}(x)$, удовлетворяющих поставленным условиям. Их разность — полином $P_n(x) = P_n^{(1)}(x) - P_n^{(2)}(x)$ — также представляет собой полином n -го порядка, который в точках сетки должен удовлетворять условиям $P_n(x_i) = 0$, $i = \overline{0, n}$, т. е. быть равным нулю в $n + 1$ точках. Такое возможно для полинома n -й степени только в случае $P_n(x) \equiv 0$. Однако это противоречит предположению о том, что $P_n^{(1)}(x) \neq P_n^{(2)}(x)$. Следовательно, решение задачи построения интерполяционного полинома единственно. ►

Замечание 4.3. Построение интерполяционного полинома вида $P(x) = a_0 + \sum_{k=1}^n a_k x_i^k$ эквивалентно решению СЛАУ с матрицей Вандермонда. Известно, что определитель Вандермонда обращается в нуль, если $x_i = x_j$, $i \neq j$. При большом числе узлов сетки $\{x_i\}$ на рассматриваемом отрезке узлы становятся близкими и, как следствие, матрица системы становится близка к вырожденной. Это может отрицательно отразиться на точности, с которой определены коэффициенты интерполяционного полинома в результате решения СЛАУ, и на свойствах полинома.

Замечание 4.4. Полиномиальная интерполяция есть частный случай приближения функции, заданной таблично, суммами вида $\sum_{i=0}^n c_i \varphi_i(x)$ при $\varphi_i(x) = x^i$, $i = \overline{0, n}$. В общей ситуации обычно рассматривается интерполяция по так называемой **чебышёвской системе функций** — системе из $n + 1$ функций $\varphi_i(x)$, $i = \overline{0, n}$, любая линейная комбинация которых не может иметь $n + 1$ различных корней на участке интерполяции. Известно, что функции $1, x, x^2, \dots, x^n$ образуют систему Чебышёва на любом отрезке.

4.2.2. Интерполяционный полином в форме Лагранжа

Интерполяционный полином единственен, но может быть записан в различных формах. Для теоретического анализа удобен **интерполяционный полином (многочлен) $L_n(x)$ в форме Лагранжа**:

$$P_n(x) = L_n(x) = \sum_{k=0}^n f_k \prod_{\substack{i=0 \\ i \neq k}}^n \frac{x - x_i}{x_k - x_i}.$$

Отметим, что многочлен n -го порядка

$$\varphi_k^n(x) = \prod_{\substack{i=0 \\ i \neq k}}^n \frac{x - x_i}{x_k - x_i}$$

обращается в нуль в точках x_j при $j \neq k$ и в единицу в точке x_k . Следовательно, полином $L_n(x)$ удовлетворяет требованию интерполяции $L_n(x_i) = f_i$. Его можно переписать с помощью функции

$$\omega(x) = \prod_{i=0}^n (x - x_i).$$

Тогда **базисный полином** представим в виде

$$\varphi_k^n(x) = \prod_{\substack{i=0 \\ i \neq k}}^n \frac{x - x_i}{x_k - x_i} = \frac{\omega(x)}{(x - x_k)\omega'(x_k)}$$

(далее верхний индекс n будем, как правило, опускать). Таким образом, полиномом Лагранжа

$$L_n(x) = \sum_{k=0}^n f_k \varphi_k(x).$$

Введем **остаточный член интерполяционного полинома**:

$$r_n(x) = f(x) - L_n(x).$$

Теорема 4.4. Пусть функция $f(x)$ имеет $(n+1)$ -ю непрерывную производную на отрезке $[a, b]$. Тогда для любого $x \in [a, b]$ и сетки $\Omega_h = \{a = x_0 < x_1 < \dots < x_n = b\}$ существует точка $\xi \in [a, b]$ такая, что остаточный член интерполяционного полинома

$$r_n(x) = \frac{1}{(n+1)!} \omega(x) f^{(n+1)}(\xi).$$

◀ Рассмотрим вспомогательную функцию

$$g(t) = f(t) - L_n(t) - r_n(x) \frac{\omega(t)}{\omega(x)},$$

где x — параметр ($x \neq x_i$, $i = \overline{0, n}$). Функция $g(t)$ имеет $n+1$ корней $t = x_i$, где $f(t) = L_n(t)$ и $\omega(t) = 0$, а также $(n+2)$ -й корень $t = x$, в котором $g(t) = g(x) = 0$ в силу определения остаточного члена $r_n(x)$. Таким образом, производная $g'_t(t)$ имеет не менее $n+1$ корней, $g''_{t^2}(t)$ — не менее n корней, \dots , $g_{t^{n+1}}^{(n+1)}(t)$ — как минимум один корень $\xi \in [a, b]$. В этой точке

$$\begin{aligned} g_{t^{n+1}}^{(n+1)}(\xi) &= f^{(n+1)}(\xi) - L_n^{(n+1)}(\xi) - r_n(x) \frac{(n+1)!}{\omega(x)} = \\ &= f^{(n+1)}(\xi) - r_n(x) \frac{(n+1)!}{\omega(x)} = 0, \end{aligned}$$

откуда

$$r_n(x) = \frac{1}{(n+1)!} \omega(x) f^{(n+1)}(\xi). \quad \blacktriangleright$$

Замечание 4.5. Если $M_{n+1} = \|f^{(n+1)}\|_C$, то оценку точности приближения можно записать в виде

$$|f(x) - L_n(x)| \leq \frac{M_{n+1}}{(n+1)!} |\omega(x)|.$$

Эта оценка справедлива как при $x \in [a, b]$ (интерполяция), так и при $x \notin [a, b]$ (экстраполяция). Однако в последнем случае точка ξ может уже находиться вне отрезка $[a, b]$ ($\xi \in [\min(a, x), \max(b, x)]$). При этом функция $f(x)$ должна иметь

$n + 1$ непрерывную производную на указанном расширенном отрезке, на котором вычисляется и M_{n+1} .

Теорема 4.5. Пусть на отрезке $[a, b]$ задана равномерная сетка $\{x_i\}$: $x_i - x_{i-1} = h$, $i = \overline{1, n}$, $x \in [a, b]$, $f(x)$ имеет непрерывную и, значит, ограниченную производную $(n + 1)$ -го порядка на $[a, b]$. Тогда

$$|f(x) - L_n(x)| \leq \frac{M_{n+1}}{n+1} h^{n+1}.$$

◀ Пусть $x = x_{i^*} + \alpha h$, $x \in [x_{i^*}, x_{i^*+1}]$, $x_{i^*} = x_0 + i^* h$, $i^* \geq 0$, $\alpha \neq 0; 1$ (в противном случае $\omega(x) = 0$). Тогда

$$\begin{aligned} \omega(x) &= \prod_{i=0}^n (x - x_i) = h^{n+1} \prod_{i=0}^n (i^* + \alpha - i) = \\ &= h^{n+1} \prod_{i=0}^{i^*} (i^* + \alpha - i) \prod_{i=i^*+1}^n (i^* + \alpha - i). \end{aligned}$$

Отсюда получим оценку

$$\begin{aligned} |\omega(x)| &= h^{n+1} \prod_{i=0}^{i^*} (i^* + \alpha - i) \prod_{i=i^*+1}^n (i - \alpha - i^*) \leqslant \\ &\leqslant h^{n+1} (i^* + 1)! (n - i^*)!. \end{aligned}$$

При этом $i^* \leq n - 1$. В результате

$$\frac{|\omega(x)|}{n!} \leq h^{n+1} \frac{(i^* + 1)!}{n(n-1)\cdots(n-i^*+1)}.$$

В числителе $i^* + 1$ сомножителей, в знаменателе $n - (n - i^*) = i^*$ сомножителей, последний из которых $n - i^* + 1 \geq 2$. Отсюда

$$\frac{|\omega(x)|}{n!} \leq h^{n+1}.$$

В результате

$$|f(x) - L_n(x)| \leq \frac{M_{n+1}}{n+1} h^{n+1}. \quad ▶$$

Теорема 4.5 дает довольно грубую оценку остаточного члена, часто можно получить более точную оценку.

Утверждение 4.1. Рассмотрим случай $n = 1$, т. е. линейную интерполяцию по отрезкам $[x_{i-1}, x_i]$. Тогда

$$|f(x) - L_n(x)| \leq \frac{M_2}{8}h^2, \quad x \in [x_{i-1}, x_i].$$

Эта оценка неулучшаема в классе функций, имеющих ограниченную вторую производную.

◀ Действительно,

$$\|r_1\|_C \leq \frac{M_2}{2!} \|(x - x_{i-1})(x - x_i)\|_C \leq \frac{M_2}{2} \frac{h^2}{4} = \frac{1}{8} M_2 h^2.$$

Для доказательства того, что оценка неулучшаема, достаточно взять в качестве примера функцию

$$f(x) = \left(x - \frac{1}{2}(x_{i-1} + x_i) \right)^2$$

с параметрами $x_{i-1} = -1$, $x_i = 1$, для которой $M_2 = 2$, $h = 2$, $L_1 \equiv 1$, $r_1 = 1$, т. е. приведенная оценка является точной. ►

Рассмотрим вопрос о погрешности экстраполяции, т. е. о вычислении $L_n(x)$ при $x \notin [a, b]$. Как показывает анализ, подобный приведенному в доказательстве теоремы 4.5:

при $x \in [b, b+h]$

$$|f - L_n| = |r_n| \leq h^{n+1} \max_{\xi \in [a, x]} |f^{(n+1)}(\xi)|;$$

при $x \in [b+h, b+2h]$

$$|f - L_n| = |r_n| \leq h^{n+1}(n+2) \max_{\xi \in [a, x]} |f^{(n+1)}(\xi)|;$$

при $x \in [b+2h, b+3h]$

$$|f - L_n| = |r_n| \leq \frac{1}{2}h^{n+1}(n+2)(n+3) \max_{\xi \in [a, x]} |f^{(n+1)}(\xi)|$$

и т. д. Из полученных оценок следует, что при удалении x от области интерполирования погрешности возрастают с такой же

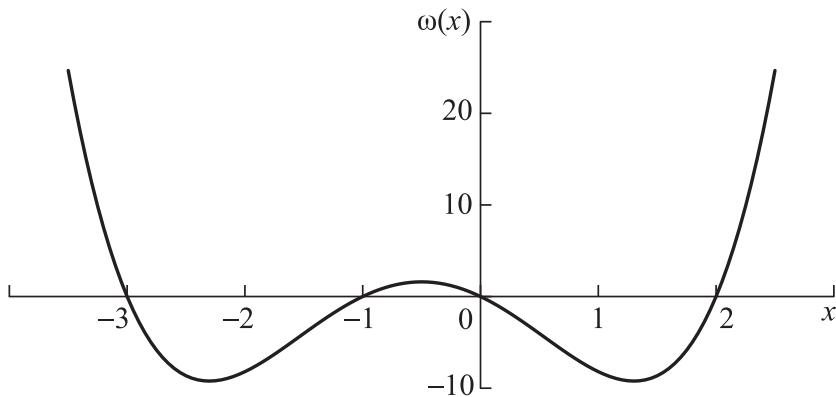


Рис. 4.3. График функции $\omega(x)$ для $n = 3$

скоростью, как и $n!$. Эта ситуация знакома по планам и прогнозам: чем на более долгий срок они составляются, тем сильнее расходятся с реальностью. Графическое пояснение представлено на рис. 4.3, где изображена функция $\omega(x)$, обращающаяся в нуль в четырех точках (т. е. $n = 3$). Вне области интерполяирования данная функция резко возрастает. Этот рост и проявляется в виде неустойчивости экстраполяции.

4.2.3. Интерполяционный полином в форме Ньютона

Для практического применения часто оказывается удобной другая форма записи интерполяционного полинома.

Определение. *Разделенными разностями* нулевого порядка называются значения $f_i = f(x_i)$ функции в точках x_i , первого порядка — значения

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

второго порядка — значения

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

и т. д. Разность k -го порядка имеет вид

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

Теорема 4.6. Справедливо равенство

$$f(x_1, \dots, x_k) = \sum_{j=1}^k \frac{f(x_j)}{\prod_{\substack{i=1 \\ i \neq j}}^k (x_j - x_i)}.$$

◀ Проведем доказательство методом математической индукции. Для $k = 2$ равенство верно. Пусть оно верно и для $k - 1 > 1$. Тогда для разности порядка k

$$\begin{aligned} f(x_1, \dots, x_{k+1}) &= \frac{1}{x_{k+1} - x_1} [f(x_2, \dots, x_{k+1}) - f(x_1, \dots, x_k)] = \\ &= \frac{1}{x_{k+1} - x_1} \left[\sum_{j=2}^{k+1} \frac{f(x_j)}{\prod_{\substack{2 \leq i \leq k+1 \\ i \neq j}} (x_j - x_i)} - \sum_{j=1}^k \frac{f(x_j)}{\prod_{\substack{1 \leq i \leq k \\ i \neq j}} (x_j - x_i)} \right]. \end{aligned}$$

Слагаемые в сумме при $j = k + 1$ дают выражение требуемого вида, входящее в предполагаемую формулу. Аналогично и при $j = 1$. При $2 \leq j \leq k$ имеем коэффициент в скобках перед $f(x_j)$, равный

$$\begin{aligned} &\frac{1}{\prod_{\substack{2 \leq i \leq k+1 \\ i \neq j}} (x_j - x_i)} - \frac{1}{\prod_{\substack{1 \leq i \leq k \\ i \neq j}} (x_j - x_i)} = \\ &= \frac{(x_j - x_1) - (x_j - x_{k+1})}{\prod_{\substack{1 \leq i \leq k+1 \\ i \neq j}} (x_j - x_i)} = \frac{x_{k+1} - x_1}{\prod_{\substack{1 \leq i \leq k+1 \\ i \neq j}} (x_j - x_i)}. \end{aligned}$$

В формуле для $f(x_1, \dots, x_{k+1})$ числитель последнего выражения сокращается со знаменателем $x_{k+1} - x_1$, стоящим перед разностью сумм. В результате получаем, что утверждение теоремы справедливо. ►

Следствие 4.1. Значение разделенной разности $f(x_1, x_2, \dots, x_k)$ не зависит от порядка следования аргументов.

Теорема 4.7. *Интерполяционный многочлен $P_n(x)$ может быть записан в форме Ньютона:*

$$\begin{aligned} P_n(x) &= f(x_0) + (x - x_0)f(x_0, x_1) + (x - x_0)(x - x_1)f(x_0, x_1, x_2) + \dots \\ &\quad \dots + (x - x_0)(x - x_1) \cdots (x - x_{n-1})f(x_0, x_1, \dots, x_n). \end{aligned}$$

◀ Проведем доказательство методом математической индукции. Обозначим через $P_i(x)$ интерполяционный полином (записанный, например, в форме Лагранжа), принимающий заданные значения в точках x_0, x_1, \dots, x_i . Для случая $i = 1$ очевидно, что

$$P_1(x) = L_1(x) = f(x_0) + (x - x_0)f(x_0, x_1).$$

Допустим, что утверждение теоремы верно при $i - 1 > 1$. Тогда многочлен $P_{i-1}(x)$ в силу единственности интерполяционного полинома может быть записан как в форме Ньютона, так и в форме Лагранжа. Представим многочлен $P_i(x)$ в виде $P_i = P_{i-1} + (P_i - P_{i-1})$. При этом оба многочлена P_i и P_{i-1} принимают одинаковые значения в точках x_0, \dots, x_{i-1} , т. е. их разность в этих точках равна нулю. Поскольку $P_i - P_{i-1}$ есть многочлен степени i , то

$$P_i - P_{i-1} = A_i(x - x_0)(x - x_1) \cdots (x - x_{i-1}).$$

Потребуем, чтобы коэффициент A_i был таким, что

$$P_i(x_i) = f(x_i) = P_{i-1}(x_i) + A_i(x_i - x_0)(x_i - x_1) \cdots (x_i - x_{i-1}).$$

Отсюда

$$\begin{aligned} A_i &= \frac{1}{\prod_{k=0}^{i-1} (x_i - x_k)} \left(f(x_i) - \sum_{j=0}^{i-1} f(x_j) \prod_{\substack{k=0 \\ k \neq j}}^{i-1} \frac{x_i - x_k}{x_j - x_k} \right) = \\ &= \frac{f(x_i)}{\prod_{k=0}^{i-1} (x_i - x_k)} - \sum_{j=0}^{i-1} \frac{f(x_j)}{\prod_{\substack{k=0 \\ k \neq j}}^{i-1} (x_j - x_k)(x_i - x_j)} = \end{aligned}$$

$$\begin{aligned}
 &= \frac{f(x_i)}{\prod_{k=0}^{i-1} (x_i - x_k)} + \sum_{j=0}^{i-1} \frac{f(x_j)}{\prod_{\substack{k=0 \\ k \neq j}}^i (x_j - x_k)} = \\
 &= \sum_{j=0}^i \frac{f(x_j)}{\prod_{\substack{k=0 \\ k \neq j}}^i (x_j - x_k)} = f(x_0, \dots, x_i).
 \end{aligned}$$

В результате $P_i = P_{i-1} + (x - x_0)(x - x_1) \cdots (x - x_{i-1})f(x_0, \dots, x_i)$, т. е. интерполяционный полином может быть записан в форме Ньютона. ►

Форма Ньютона интерполяционного полинома удобна для численных расчетов, в особенности выполняемых вручную. Например, она позволяет с меньшими трудозатратами увеличить или уменьшить число узлов сетки, которые используются для построения интерполяционного полинома.

Для построения полинома в форме Ньютона на практике используют таблицу, приведенную на рис. 4.4. По каждой паре соседних узлов вычисляют разделенные разности первого порядка. Таких разностей будет n . По каждой паре соседних разделенных разностей первого порядка вычисляют разделенную разность второго порядка; их будет $n - 1$. Продолжая этот процесс, доходят

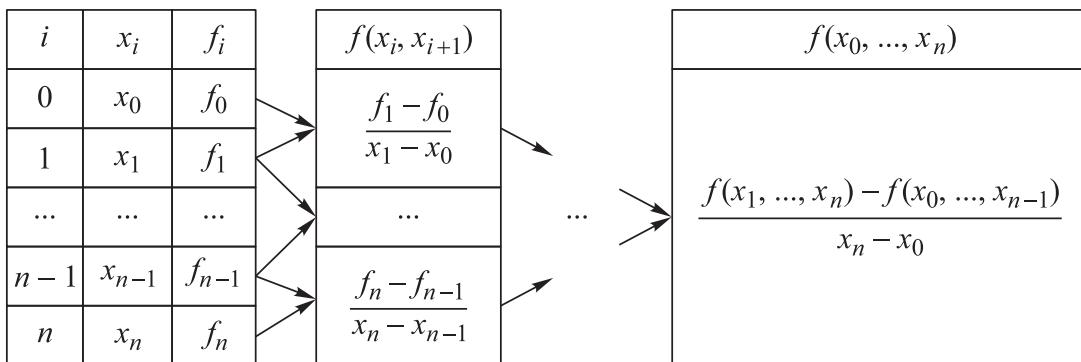


Рис. 4.4. Схема вычисления разделенных разностей для построения интерполяционного полинома в форме Ньютона

до единственной разделенной разности n -го порядка. Таким образом, таблица разделенных разностей будет иметь треугольный вид. При этом для записи интерполяционного полинома в форме Ньютона понадобятся лишь верхние строки каждого блока схемы.

Пример 4.4. Проведем глобальную полиномиальную интерполяцию функции из примера 4.2. Для построения интерполяционного полинома в форме Ньютона удобно использовать приведенную на рис. 4.5 схему.

i	x_i	f_i	$f(x_i, x_{i+1})$	$f(x_i, x_{i+1}, x_{i+2})$
0	2	20	$\frac{20 - 12}{6 - 2} = -\frac{8}{4} = -2$	
1	6	12	$\frac{11 - 12}{11 - 6} = -\frac{1}{5}$	$\frac{-1/5 + 2}{9} = \frac{1}{5}$
2	11	11		

Рис. 4.5. Схема вычисления разделенных разностей для функции из примеров 4.2 и 4.4

Исходя из данных этой схемы, получаем интерполяционный полином в форме Ньютона

$$L_2(x) = 20 + (-2)(x - 2) + \frac{1}{5}(x - 2)(x - 6) = \frac{1}{5}x^2 - \frac{18}{5}x + \frac{132}{5}.$$

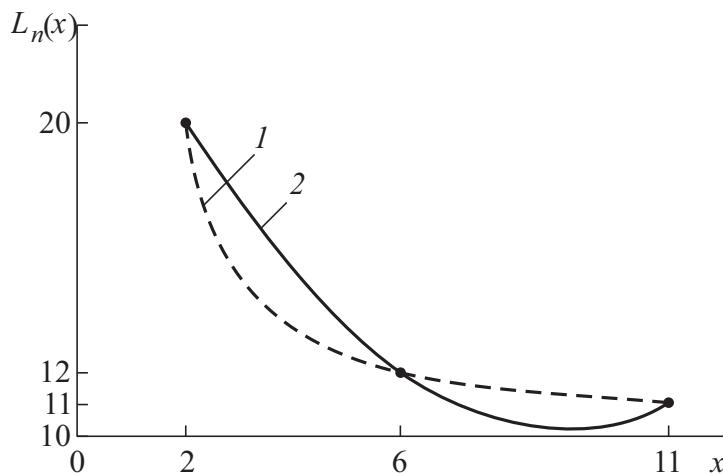


Рис. 4.6. Реконструкции функции из примеров 4.2 и 4.4:
1 — дробно-линейная; 2 — полиномиальная

На рис. 4.6 представлен полученный полином (кривая 2) и дробно-линейная интерполяция (кривая 1) исследуемой функции. •

Замечание 4.6. На практике часто требуется вычислить обратную функцию. При этом даже если известно явное выражение функции $y = f(x)$, не всегда можно найти явное выражение для обратной функции $x = f^{-1}(y)$. Однако численно определить обратную функцию гораздо легче. Для этого выберем достаточно мелкую сетку $\{x_i\}$ и вычислим значения $f_i = f(x_i)$. В результате получим таблично заданную функцию (x_i, f_i) . Поменяв местами столбцы этой таблицы и рассмотрев $\{f_i\}$ как аргумент, а $\{x_i\}$ как функцию, получим таблицу для обратной функции. Таким образом, обратная функция табулирована, в узлах $\{f_i\}$ она известна. Промежуточные значения $x(f)$ можно найти с помощью интерполирования. Такой прием называется **обратной интерполяцией**.

Отметим, что при использовании описанного алгоритма возникают проблемы, если функция $f(x)$ немонотонна. Это следует из общей теории математического анализа. Однако и в этом случае можно разбить отрезок, на котором строится обратная функция, на такие подотрезки, чтобы на каждом из них функция была монотонна.

Метод обратной интерполяции можно применять для решения нелинейных уравнений, при этом задачу $f(x) = 0$ заменяют задачей восстановления функции $\tilde{x}(f)$, тогда корнем уравнения будет $x = \tilde{x}(0)$.

4.2.4. Интерполяционный полином Эрмита

Рассмотрим расширенную постановку задачи интерполирования. Пусть есть $m + 1$ точек

$$a = x_0 < x_1 < \dots < x_m = b$$

и в каждом k -м узле сетки x_k известно n_k значений функции и ее производных $f^{(0)}(x_k), \dots, f^{(n_k-1)}(x_k)$, где n_k — кратность k -го узла. Построим интерполяционный многочлен, проходящий через эти точки так, что в каждой точке x_k он и его производные порядков $1, 2, \dots, n_k - 1$ принимают заданные значения. Будем строить многочлен степени $n = \sum_{k=0}^m n_k - 1$.

Определение. Многочлен $H_n(x)$, удовлетворяющий условиям

$$H_n^{(i)}(x_k) = f^{(i)}(x_k), \quad i = \overline{0, n_k - 1}; \quad k = \overline{0, m},$$

называется *интерполяционным многочленом Эрмита*, а интерполяция — *интерполяцией с кратными узлами (интерполяцией Эрмита)*.

Можно доказать, что такой многочлен $H_n(x)$ существует и является единственным (при несовпадающих точках сетки).

Коэффициенты многочлена $H_n(x)$ линейно выражены через $f^{(i)}(x_k)$, и его можно искать в виде

$$H_n(x) = \sum_{k=0}^m \sum_{i=0}^{n_k-1} c_{i,k}(x) f^{(i)}(x_k),$$

где $c_{i,k}(x)$ — полиномы n -го порядка. Их можно получить в явном виде по аналогии с полиномом Лагранжа (Ньютона).

Проще найти коэффициенты полинома Эрмита, воспользовавшись формой Ньютона интерполяционного полинома: интерполяцию с кратными узлами можно представить как построение интерполяционного полинома по $n + 1$ точкам, в которых каждая точка x_k повторяется n_k раз, т. е. количество повторов соответствует ее кратности. Вычисление разделенных разностей по представленным формулам ведет к неопределенностям типа «0/0». Однако возникшие неопределенностии могут быть раскрыты согласно правилу Лопитала. При этом появляются производные.

Этим замечанием относительно общей ситуации и ограничимся. Часто оказывается, что в конкретном случае проще решить задачу, исходя из ее конкретной постановки, а не пользоваться готовой формулой. Приведем примеры.

Пример 4.5. Построим полином Эрмита на сетке из двух точек $\Omega_h = x_0, x_1$, в каждой из которых заданы значения непрерывно дифференцируемой функции f и ее производных

$$f_0 = f(x_0); \quad f_1 = f(x_1); \quad f'_0 = f'(x_0); \quad f'_1 = f'(x_1).$$

Как и в примере 4.4, составим схему для вычисления разделенных разностей (рис. 4.7). При этом при вычислении разделенных разностей вида $\frac{f_0 - f_0}{x_0 - x_0}$ будем использовать разложение разности $f(x') - f(x'')$ по формуле Тейлора при $x' \rightarrow x_0$ и $x'' \rightarrow x_0$, а именно

$$\begin{aligned} f(x') - f(x'') &= f(x_0) + f'(x_0)(x' - x_0) + O((x' - x_0)^2) - \\ &\quad - f(x_0) - f'(x_0)(x'' - x_0) + O((x'' - x_0)^2) = \\ &= f'(x_0)(x' - x'') + O((x' - x_0)^2) + O((x'' - x_0)^2). \end{aligned}$$

Тогда предел

$$\lim_{\substack{x' \rightarrow x_0 \\ x'' \rightarrow x_0}} \frac{f(x') - f(x'')}{x' - x''} = f'(x_0) = f'_0.$$

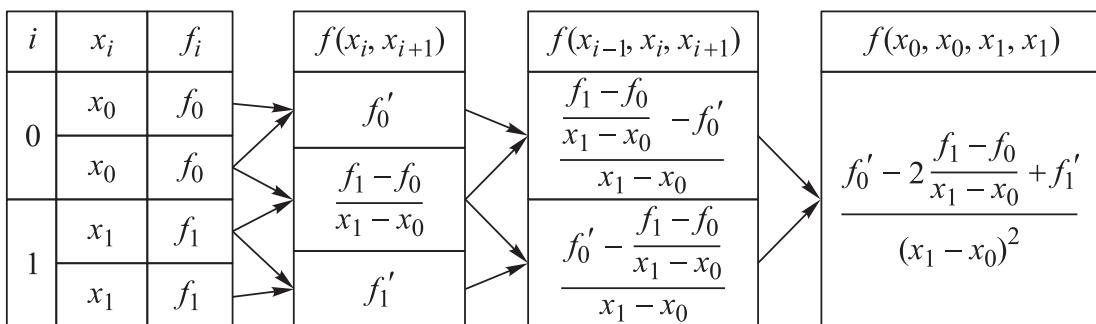


Рис. 4.7. Схема вычисления разделенных разностей для полинома Эрмита

Запишем интерполяционный полином, пользуясь формой Ньютона:

$$\begin{aligned} H_1(x) = f_0 + (x - x_0)f'_0 + \frac{(x - x_0)^2}{x_1 - x_0} \left(\frac{f_1 - f_0}{x_1 - x_0} - f'_0 \right) + \\ + \frac{(x - x_0)^2(x - x_1)}{(x_1 - x_0)^2} \left(f'_0 - 2 \frac{f_1 - f_0}{x_1 - x_0} + f'_1 \right). \end{aligned}$$

Теперь перегруппируем слагаемые и запишем коэффициенты при значениях f_0, f_1, f'_0, f'_1 :

$$\begin{aligned} f_0: & \quad 1 - \frac{(x - x_0)^2}{(x_1 - x_0)^2} + 2 \frac{(x - x_0)^2(x - x_1)}{(x_1 - x_0)^3}; \\ f_1: & \quad \frac{(x - x_0)^2}{(x_1 - x_0)^2} - 2 \frac{(x - x_0)^2(x - x_1)}{(x_1 - x_0)^3}; \\ f'_0: & \quad x - x_0 - \frac{(x - x_0)^2}{x_1 - x_0} + \frac{(x - x_0)^2(x - x_1)}{(x_1 - x_0)^2}; \\ f'_1: & \quad \frac{(x - x_0)^2(x - x_1)}{(x_1 - x_0)^2}. \end{aligned}$$

Коэффициент при f'_1 имеет вид $(x - x_1)\varphi_1^2(x)$, где $\varphi_1(x)$ – базисная функция лагранжевой интерполяции для узла x_1 .

Аналогично для коэффициента при f'_0 получим

$$\begin{aligned} x - x_0 - \frac{(x - x_0)^2}{x_1 - x_0} + \frac{(x - x_0)^2(x - x_1)}{(x_1 - x_0)^2} = \\ = \frac{(x - x_0)(x_1 - x_0) - (x - x_0)^2}{x_1 - x_0} + \frac{(x - x_0)^2(x - x_1)}{(x_1 - x_0)^2} = \\ = -\frac{(x - x_0)(x - x_1)}{x_1 - x_0} + \frac{(x - x_0)^2(x - x_1)}{(x_1 - x_0)^2} = \\ = \frac{(x - x_0)(x - x_1)^2}{(x_1 - x_0)^2} \end{aligned}$$

или, с использованием базисных функций лагранжевой интерполяции, $(x - x_0)\varphi_0^2(x)$.

Таким образом, получили следующий вид базисных полиномов:

$$c_{1,k} = (x - x_k) \varphi_k^2(x), \quad k = 0; 1.$$

Выполнив аналогичные по смыслу преобразования, имеем

$$c_{0,0} = \left(1 + (x - x_0) \frac{2}{x_1 - x_0} \right) \varphi_0^2(x);$$

$$c_{0,1} = \left(1 - (x - x_1) \frac{2}{x_1 - x_0} \right) \varphi_1^2(x).$$

Далее получим более общие формулы для этих полиномов. •

Пример 4.6. Рассмотрим теперь следующую задачу: требуется построить полином Эрмита по данным в $m+1$ точках

$$a = x_0 < x_1 < \dots < x_m = b,$$

в каждой из которых известны значения функции и ее производной (здесь $n_k = 2$):

$$f(x_k) = f_k; \quad f'(x_k) = f'_k, \quad k = \overline{0, m}.$$

В этом случае степень интерполяционного полинома $n = 2(m+1) - 1 = 2m + 1$. Построим полином в явном виде. Поскольку коэффициенты интерполяционного полинома линейно выражены через заданные значения функции и ее производной в точках сетки, то полином Эрмита можно записать в следующем виде:

$$H_n(x) = \sum_{k=0}^m (a_k(x)f_k + b_k(x)f'_k),$$

где $a_k(x)$, $b_k(x)$ — полиномы степени n . Сопоставление полинома указанного вида и условий, налагаемых на него в точках сетки, дает следующие уравнения для определения коэффициентов полиномов $a_k(x)$, $b_k(x)$, $k = \overline{0, m}$:

$$a_k(x_i) = \delta_{ki}; \quad a'_k(x_i) = 0; \quad b_k(x_i) = 0; \quad b'_k(x_i) = \delta_{ki}, \quad i = \overline{0, m}.$$

Каждый полином содержит $n + 1 = 2m + 2$ коэффициентов, которые требуется определить. Число решаемых уравнений равно числу неизвестных, так что задача может быть разрешимой.

Для построения полиномов в явном виде, как и в примере 4.5, применим базисные полиномы Лагранжа $\varphi_k^m(x)$, удовлетворяющие условиям $\varphi_k^m(x_i) = \delta_{ki}$. По аналогии со случаем двух узлов запишем $b_k(x)$ в виде

$$b_k(x) = (x - x_k) (\varphi_k^m(x))^2.$$

Действительно, первое условие $b_k(x_i) = 0$ выполнено. Выполнение второго станет очевидным, если записать

$$b'_k(x) = (\varphi_k^m(x))^2 + 2(x - x_k)\varphi_k^m(x)(\varphi_k^m(x))'.$$

Несколько сложнее предсказать вид второго полинома $a_k(x)$. Анализ примера 4.5 и накладываемых на базисный полином $a_k(x)$ условий показывает, что его следует искать в форме

$$a_k(x) = [1 + \alpha_k(x - x_k)] (\varphi_k^m(x))^2,$$

где α_k — пока не определенный коэффициент.

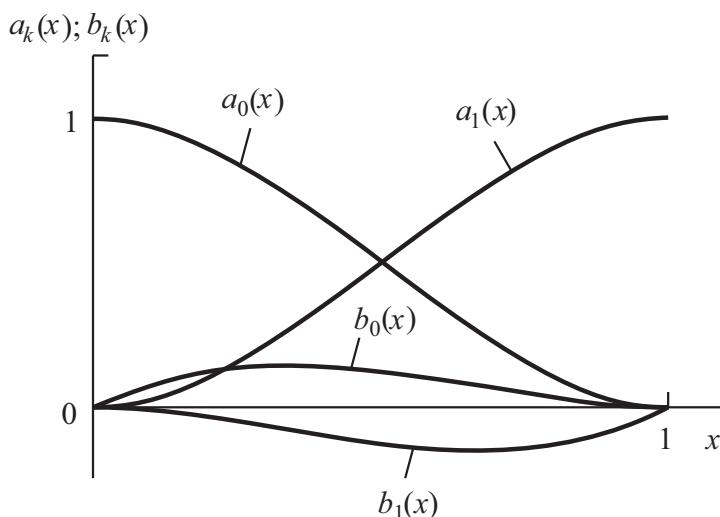


Рис. 4.8. Базисные функции эрмитовой интерполяции для случая сетки из двух точек

Легко видеть, что первое условие, которому должен удовлетворять искомый полином, выполнено. Второе условие позволяет найти единственный неизвестный коэффициент. В результате имеем следующее решение:

$$a_k(x) = \left[1 - 2(\varphi_k^m)'(x_k)(x - x_k) \right] (\varphi_k^m(x))^2.$$

Искомый полином Эрмита построен. В качестве примера на рис. 4.8 приведем базисные функции $a_k(x)$ и $b_k(x)$ для $m = 1$, т. е. для двух точек сетки. •

Пример 4.7. Рассмотрим «несимметричную» задачу: требуется построить полином Эрмита по заданным значениям функции в $m + 1$ точках

$$a = x_0 < x_1 < \dots < x_m = b.$$

При этом в одной точке (x_{k_0}) известно значение ее производной:

$$f(x_k) = f_k, \quad k = \overline{0, m};$$

$$f'(x_{k_0}) = f'_{k_0}.$$

В данном примере $n_k = 1$ при $k \neq k_0$; $n_{k_0} = 2$. Отсюда $n = (m + 1) + 1 - 1 = m + 1$.

Решение задачи также нетрудно получить явно. Полином Эрмита можно записать в следующем виде:

$$H_n(x) = b_{k_0}(x)f'_{k_0} + \sum_{k=0}^m a_k(x)f_k,$$

где $a_k(x)$, $k = \overline{0, m}$, и $b_{k_0}(x)$ — полиномы степени n .

Потребуем выполнения условий в точках сетки x_i , $i = \overline{0, m}$,

$$a_k(x_i) = \delta_{ki}; \quad b_{k_0}(x_i) = 0,$$

а в точке x_{k_0}

$$a'_k(x_{k_0}) = 0; \quad b'_{k_0}(x_{k_0}) = 1.$$

Опуская рассуждения, аналогичные приведенным при рассмотрении примера 4.6, запишем окончательное решение:

$$b_{k_0}(x) = (x - x_{k_0})\varphi_{k_0}^m(x); \quad a_k(x) = [1 + \alpha_k(x - x_k)]\varphi_k^m(x),$$

$$\text{где } \alpha_k = -\frac{(\varphi_k^m)'(x_{k_0})}{\delta_{kk_0} + (\varphi_k^m)'(x_{k_0})(x_{k_0} - x_k)}.$$

Требуемый полином Эрмита построен. •

Пример 4.8. Простейшим примером полинома Эрмита является полином Тейлора, построенный по заданным в одной точке значениям функции и ее производных до n -го порядка включительно. Он имеет вид

$$\mathcal{T}_n(x) = \sum_{k=0}^n \frac{1}{k!} f^{(k)}(x_0)(x - x_0)^k.$$

По сути, полином Тейлора не интерполянт, а экстраполянт. Известно, что разность значений $n+1$ раз непрерывно дифференцируемой в некоторой окрестности точки x_0 функции $f(x)$ и ее полинома Тейлора $\mathcal{T}_n(x)$ в указанной окрестности может быть представлена через производную функции $(n+1)$ -го порядка:

$$f(x) - \mathcal{T}_n(x) = \frac{1}{(n+1)!} f^{(n+1)}(\xi)(x - x_0)^{n+1},$$

где $\xi \in (x_0, x)$ (или $\xi \in (x, x_0)$). Как видно, последнее выражение погрешности сходно с выражением для погрешности интерполяции, полученным в теореме 4.4. •

4.3. Сходимость и устойчивость полиномиальной интерполяции

4.3.1. Оптимизация узлов сетки

При проведении интерполяции желательно, чтобы была достигнута минимальная погрешность $\|f - \tilde{f}\|$ в некоторой норме и чтобы норма погрешности стремилась к нулю с увеличением

числа точек интерполяции. В этом случае говорят, что *интерполяционный процесс сходится*. Для описания сходимости этого процесса используют те же характеристики, что и для описания сходимости функциональных последовательностей.

В случае полиномиальной аппроксимации $\tilde{f}(x) = L_n(x)$ и

$$\|f - \tilde{f}\|_C = \|f - L_n\|_C = \|r_n\|_C \leq \frac{M_{n+1}}{(n+1)!} \|\omega\|_C.$$

Поставим задачу выбора такого набора узлов, чтобы равномерная (в пространстве C) норма правой части данного неравенства была минимальна, т. е. будем искать

$$\min_{x_0, \dots, x_n} \max_{[a,b]} |\omega(x)|, \quad \omega(x) = \prod_{i=0}^n (x - x_i).$$

Это классическая задача о построении **полинома** $(n+1)$ -го порядка, **наименее уклоняющегося от нуля**. В данном случае коэффициент при старшей степени неизвестной равен единице. Несмотря на это различие, решение задачи совпадает с решением, которое приведено при построении метода Ричардсона (см. 3.1.2). Эта задача решена В.А. Марковым. Искомый полином представляет собой полином Чебышёва первого рода и имеет вид

$$\omega(x) = T_{n+1}(x) = \frac{(b-a)^{n+1}}{2^{2n+1}} \cos \left((n+1) \arccos \frac{2x-(b+a)}{b-a} \right).$$

Корни полинома задаются следующим соотношением:

$$x_k = \frac{a+b}{2} + \frac{b-a}{2} \cos \frac{(2k+1)\pi}{2(n+1)}, \quad k = \overline{0, n},$$

они являются узлами интерполяции. Для такого набора узлов

$$\|\omega\|_C = \frac{1}{2^{2n+1}} (b-a)^{n+1};$$

$$\|f - L_n\|_C \leq \frac{M_{n+1}}{(n+1)!} \frac{(b-a)^{n+1}}{2^{2n+1}}.$$

Полученная оценка называется ***наилучшей равномерной оценкой погрешности интерполяции*** и является неулучшаемой. Для доказательства ее неулучшаемости достаточно указать такую функцию, для которой в полученной оценке погрешности реализуется равенство.

Рассмотрим интерполяцию на чебышёвской сетке из $n + 1$ узлов функции

$$f(x) = a_{n+1}x^{n+1} + a_nx^n + \dots + a_0, \quad a_{n+1} > 0,$$

с помощью интерполяционного полинома n -й степени. Тогда $M_{n+1} = \|f^{(n+1)}\|_C = a_{n+1}(n+1)!$. В результате имеем следующее точное представление остаточного члена интерполяционного полинома:

$$f(x) - L_n(x) = \frac{1}{(n+1)!} \omega(x) f^{(n+1)}(\xi) = |a_{n+1}| \omega(x) = |a_{n+1}| T_{n+1}(x).$$

Норма этого остаточного члена в точности совпадает с наилучшей равномерной оценкой погрешности интерполяции, что доказывает неулучшаемость полученной оценки.

С учетом формулы Стирлинга ($n! \approx n^n e^{-n} \sqrt{2\pi n}$) наилучшая оценка погрешности может быть преобразована к виду

$$\frac{M_{n+1}}{(n+1)!} \frac{(b-a)^{n+1}}{2^{2n+1}} \approx M_{n+1} \frac{e^{n+1}}{\sqrt{2\pi(n+1)}} \left(\frac{b-a}{n+1}\right)^{n+1} \frac{1}{2 \cdot 4^n}.$$

Если же интерполяция проводится на равномерной сетке, то, согласно теореме 4.5,

$$|f - L_n| \leq \frac{M_{n+1}}{n+1} h^{n+1} = \frac{M_{n+1}}{n+1} \left(\frac{b-a}{n}\right)^{n+1}.$$

Соответственно, оценка в случае интерполяции по узлам многочлена Чебышёва отличается от оценки при использовании равномерной сетки в следующее число раз:

$$\sqrt{\frac{n+1}{2\pi}} \left(\frac{e}{4}\right)^n \frac{1}{2} \left(1 - \frac{1}{n+1}\right)^{n+1} \approx \sqrt{\frac{n+1}{2\pi}} \left(\frac{e}{4}\right)^n \frac{1}{2}.$$

Для небольших n эта величина не очень мала. Но чебышёвские сетки обладают и другими преимуществами по сравнению с равномерными.

4.3.2. Устойчивость интерполяционного полинома относительно погрешностей функции

Пусть значения функции f известны не точно, а лишь с некоторой погрешностью δf_i : $f_i = f_i^0 + \delta f_i$. Возникает вопрос, как сильно исказится при этом интерполяционный полином?

Интерполяционный полином линеен по значениям функции f_i . В случае их неточного задания интерполянт

$$L_n(x) = L_n(x, f^0 + \delta f) = L_n(x, f^0) + L_n(x, \delta f).$$

В данной форме записи в правой части явно указана зависимость интерполяционного полинома от значений функции.

Для того чтобы установить влияние погрешности входных данных на построенный полином, необходимо оценить $\max_{\|\delta f\| \leq \delta} \|L_n(x, \delta f)\|_C$.

Если ввести нормированную погрешность $\delta f_i = \delta \widetilde{\delta f}_i$, то норма возмущения $\widetilde{\delta f}_i$ будет не больше единицы. Тогда для оценки влияния погрешности входных данных требуется вычислить величину η , равную $\max_{\|\delta f\| \leq 1} \|L_n(x, \widetilde{\delta f})\|_C$. В этом случае оценка возмущения интерполянта по множеству $\|\delta f\| \leq \delta$ есть $\delta \eta$ в силу линейности L_n по значениям функции.

Определение. Величина $\eta = \max_{\|\delta f\| \leq 1} \|L_n(x, \widetilde{\delta f})\|_C$ называется **константой Лебега** или **нормой интерполяционного полинома** на данной сетке (так как $\|L_n\| \leq \eta \|f\|$).

Константа Лебега η представляет собой оценку чувствительности интерполяционного полинома к погрешностям в задании f_i

и определяется прежде всего расположением узлов сетки. Приведем без доказательства важнейшие результаты:

для равномерной сетки

$$\eta = O(2^n);$$

для чебышёвской сетки

$$\eta = O(\ln n).$$

Таким образом, погрешности, связанные с неточностью входной информации, на равномерной сетке возрастают так же быстро, как 2^n . Поэтому на практике равномерные сетки при полиномиальной интерполяции используются, как правило, лишь при $n \leq 5$ ($2^5 = 32$).

4.3.3. Устойчивость интерполяционного полинома относительно априорной информации

Пусть для функции $y = f(x)$ построен полином $L_n(x)$ на некоторой сетке. Что будет, если в действительности $f(x)$ не имеет $(n+1)$ -й ограниченной производной?

Еще С.Н. Бернштейн доказал, что полиномы, интерполирующие функцию $y = f(x) = |x|$ (только липшиц-непрерывную функцию) на равномерной сетке на отрезке $[-1; 1]$, таковы, что

$$\|f - L_n\|_C \rightarrow \infty, \quad n \rightarrow \infty.$$

При этом $L_n(x)$ не сходится к $f(x)$ ни в одной точке отрезка $[-1; 1]$ за исключением точек $\{-1; 0; 1\}$.

В общем случае верна следующая теорема (приведем ее без доказательства).

Теорема 4.8 (теорема Фабера). Для любой последовательности сеток $\{\Omega_h\}$, $\Omega_h \subset [a, b]$, существует непрерывная на отрезке $[a, b]$ функция $f(x)$ такая, что

$$\{L_n(x)\} \not\rightarrow f(x),$$

т. е. последовательность полиномов $L_n(x)$ не сходится к $f(x)$ равномерно на $[a, b]$.

Однако справедливо и обратное утверждение — теорема Марцинкевича, приведем ее также без доказательства.

Теорема 4.9 (теорема Марцинкевича). Для любой непрерывной на отрезке $[a, b]$ функции $f(x)$ существует последовательность сеток $\{\Omega_h\}$ такая, что

$$\{L_n(x)\} \rightrightarrows f(x),$$

т. е. последовательность полиномов $L_n(x)$ сходится к $f(x)$ равномерно на $[a, b]$.

Такие сетки необходимо строить для каждой функции отдельно. Процесс их построения очень сложен и в практических расчетах не используется.

Теория интерполяции — глубоко и хорошо разработанный раздел математики. В частности, доказаны существование и единственность интерполяционного полинома P_n , реализующего наилучшее равномерное приближение

$$\min_{P_n} \|P_n - f\|_C = E_n(f).$$

Пример 4.9. Рассмотрим задачу построения наилучшего равномерного приближения функции $f(x) = x^p$ при $p > 0$ на отрезке $[0; 1]$ полиномом нулевого порядка $P_0(x) = a$.

Легко видеть, что при $a \in [0; 1]$

$$\|P_0 - f\|_C = \max(a, 1 - a).$$

Если искомое a не принадлежит отрезку $[0; 1]$, то норма погрешности заведомо больше единицы. Найдем такое a , при котором рассматриваемая погрешность $\|P_0 - f\|_C$ минимальна. Очевидно, что $a = 0,5$. •

Таким образом, это задача поиска минимального значения максимума (задача о минимаксе). Такая задача часто решается при анализе различных методов вычислений. Приведенный пример и способ его решения можно обобщить и на более общую задачу реконструкции функции нескольких переменных.

Пример 4.10. Рассмотрим непрерывную на области задания своих аргументов функцию $u = f(x_1, \dots, x_n)$. Пусть ее аргументы заданы приближенно: $x_i = x_i^* \pm \Delta(x_i^*)$, $i = \overline{1, n}$. Как известно из теории погрешностей, это означает, что

$$x_i \in [x_i^* - \Delta(x_i^*); x_i^* + \Delta(x_i^*)], \quad i = \overline{1, n}.$$

Следовательно, функция рассматривается на n -мерном параллелепипеде G , стороны которого определяются точностью задания компонент аргумента.

Требуется найти постоянную u^* , которая бы наилучшим в смысле равномерной нормы образом приближала данную функцию на G . Иными словами, необходимо найти значение u^* , наилучшим образом приближающее значение функции u в случае неточного задания аргументов. Казалось бы, решением является значение

$$u^{**} = f(x_1^*, \dots, x_n^*).$$

Однако это не так (рис. 4.9).

Поскольку функция непрерывна на замкнутом параллелепипеде G , она достигает на нем своих точных нижней и верхней граней:

$$\begin{aligned} u_1 &= \inf_{(x_1, \dots, x_n) \in G} f(x_1, \dots, x_n); \\ u_2 &= \sup_{(x_1, \dots, x_n) \in G} f(x_1, \dots, x_n). \end{aligned}$$

В результате решаемая задача построения u^* преобразуется к следующему виду:

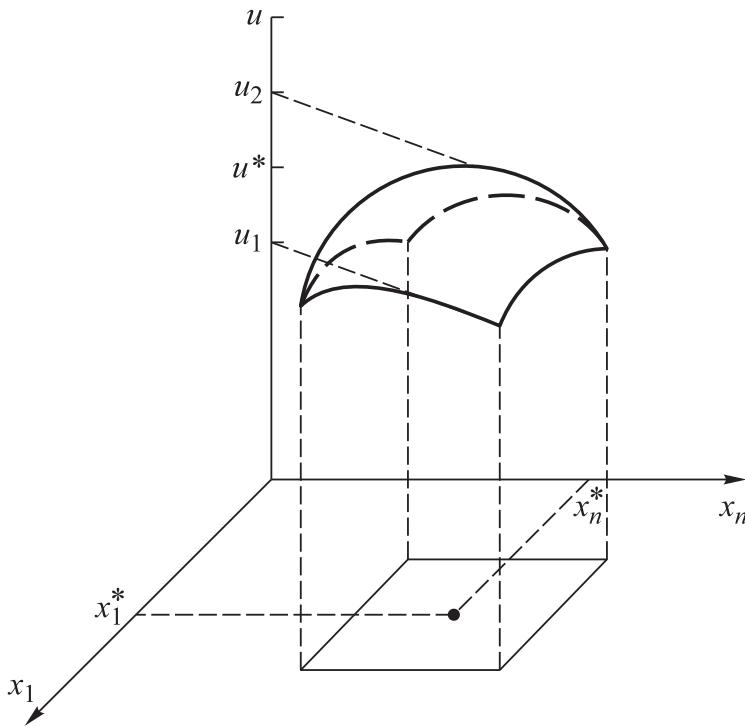


Рис. 4.9. Геометрическая интерпретация приближения функции, аргументы которой заданы неточно

$$\begin{aligned} \min_{u^*} \|u^* - f\|_C &= \min_{u^*} \max_{(x_1, \dots, x_n) \in G} |u^* - f(x_1, \dots, x_n)| = \\ &= \min_{u^*} \max_{u \in [u_1, u_2]} |u^* - u| = \min_{u^*} \max_{u^*} \{|u^* - u_1|, |u^* - u_2|\}. \end{aligned}$$

Из последнего выражения легко найти u^* , поскольку очевидно, что $u_1 \leq u^* \leq u_2$, следовательно, минимум достигается при u^* , удовлетворяющем равенству

$$u^* - u_1 = u_2 - u^*.$$

Отсюда получаем, что $u^* = (u_1 + u_2)/2$. •

Вернемся к задаче построения наилучшего равномерного приближения в общем случае. Алгоритм построения такого P_n известен, но он очень сложен и на практике не применяется. Однако известно, в частности, что асимптотика $E_n(f)$ при больших n однозначно связана с гладкостью функции f .

Если f имеет на отрезке $[a, b]$ ограниченную l -ю производную, то при $n \rightarrow \infty$ погрешность полинома наилучшего приближения

$$E_n(f) \sim O\left(n^{-l}\right).$$

Необычность ситуации заключается в том, что стандартная (на равномерной сетке, содержащей $n + 1$ точек) оценка отклонения $L_n(x)$ от $f(x)$ зависит от нормы $(n + 1)$ -й производной. Эта оценка вида

$$|r_n| \leq \frac{M_{n+1}}{n+1} h^{n+1}$$

неприменима при фиксированной гладкости функции и $n \rightarrow \infty$. Но даже при наличии производных любого порядка сходимость зависит от соотношения между M_{n+1} и h^{n+1} . Известны примеры функций $f \in C^\infty[a, b]$, для которых $|r_n|$ не сходится к нулю при $n \rightarrow \infty$.

Пример 4.11. Рассмотрим *пример Рунге* — функцию

$$f(x) = \frac{1}{1 + 25x^2}.$$

Использование глобальной полиномиальной интерполяции на равномерной сетке дает расходимость на участках $|x| \in (0,73; 1)$ при бесконечном увеличении числа точек разбиения. Причиной этого, очевидно, является увеличение нормы производной данной функции при возрастании ее порядка. ●

Таким образом, лишь для полинома наилучшего приближения заведомо можно получить

$$r_n \sim O\left(n^{-l}\right),$$

где l фиксировано.

Замечание 4.7. Сходимость интерполянта на равномерной сетке к исходной функции при наличии у нее гладкости, обеспечивающей оценку погрешности интерполяции $|r_n|$, в слу-

чае фиксированного числа точек при уменьшении шага сетки гарантируется оценкой $r_n \sim O(h^l)$. Однако при такой оценке предполагается возможность измельчать сетку, на которой заданы значения исходной функции, в непосредственной окрестности произвольной точки. Ясно, что это далеко не всегда возможно.

Замечание 4.8. Рассмотрим фиксированный участок, на котором заданы точки интерполяции, и функцию, имеющую производные любого порядка. Пусть число точек может стремиться к бесконечности, причем точки расположены произвольным образом. Обеспечивают ли указанные условия сходимость интерполянта? Ответ, вообще говоря, отрицательный. Для пояснения можно привести пример функции

$$f(x) = \begin{cases} 0, & x \in [-1; 0], \\ \exp(-1/x), & x \in (0; 1]. \end{cases}$$

Она имеет производные всех порядков. Однако если расположить точки интерполяции на левой половине отрезка $[-1; 1]$, то интерполянт будет тождественно равен нулю. Погрешность будет фиксирована и не будет зависеть от числа точек.

Несмотря на сложность построения полинома наилучшего приближения, вопрос о существовании полинома, дающего отклонение от f , близкое к оптимальному, имеет довольно простой ответ.

Теорема 4.10. Пусть $L_n(x)$ — интерполяционный полином на чебышёвской сетке для функции $f(x)$. Тогда

$$\|L_n - f\|_C \leq (1 + \tilde{C} \ln n) E_n(f).$$

◀ Запишем соотношение

$$f_i = f(x_i) = P_n(x_i) + f(x_i) - P_n(x_i),$$

где $P_n(x)$ — полином наилучшего приближения.

В силу линейности полинома L_n по значениям функции имеем

$$\begin{aligned} L_n(x) &= L_n(x, f_i) = L_n(x, P_n(x_i)) + L_n(x, f_i - P_n(x_i)) = \\ &= P_n(x) + L_n(x, f_i - P_n(x_i)). \end{aligned}$$

Равенство $L_n(x, P_n(x_i)) = P_n(x)$ справедливо вследствие единственности интерполяционного полинома. Для оценки второго слагаемого используем неравенство

$$\|L_n(x, f_i - P_n(x_i))\|_C \leq \eta \|f - P_n\|_C \leq \tilde{C} \ln n E_n(f).$$

Следовательно,

$$\begin{aligned} \|L_n - f\|_C &= \|P_n - f + L_n(x, f_i - P_n(x_i))\|_C \leq \\ &\leq E_n(f) + \tilde{C} \ln n E_n(f) = (1 + \tilde{C} \ln n) E_n(f). \quad \blacktriangleright \end{aligned}$$

Таким образом, интерполяционный полином на чебышёвской сетке вследствие относительно медленного возрастания функции $\ln n$ почти наилучший. В случае функции, имеющей l ограниченных производных ($l \geq 1$), с гарантией выполняется сходимость $\|L_n - f\|_C \rightarrow 0$ при $n \rightarrow \infty$.

4.3.4. Наилучшие приближения в гильбертовом пространстве

Ранее были рассмотрены в основном задачи об интерполяции функции, заданной таблично. Представим более подробно общую задачу об аппроксимации функции $f(x)$, являющейся элементом некоторого линейного нормированного пространства H . Рассмотрим в нем линейно независимую систему функций $\{\varphi_i(x)\}$, $i = \overline{0, n}$.

Определение. Функция $\varphi(x) = \sum_{i=0}^n c_i \varphi_i(x)$ называется **обобщенным полиномом**, построенным по системе $\{\varphi_i(x)\}$, $i = \overline{0, n}$.

Найдем такой обобщенный полином, который дает минимальное отклонение от функции $f(x)$.

Определение. Обобщенный полином $\tilde{\varphi}(x)$, являющийся решением задачи минимизации $\min_{\varphi} \|f - \varphi\|_H$, называется **элементом наилучшего приближения**.

Существование и единственность элемента наилучшего приближения определяются пространством H , которому принадлежат рассматриваемые функции.

Приведем пример пространства $L_1[-1; 1]$, в котором проводится приближение функции $f(x) \equiv 1$ прямой $\varphi = cx$. Тогда имеет место равенство

$$\|f - \varphi\|_{L_1} = \begin{cases} 2, & c \in [-1; 1], \\ \frac{c^2 + 1}{|c|} > 2, & c \in (-\infty, -1) \cup (1, +\infty). \end{cases}$$

Таким образом, любая функция $\varphi = cx$ при $c \in [-1; 1]$ есть наилучшее приближение тождественно равной единице функции в смысле пространства $L_1[-1; 1]$.

Рассмотрим случай вещественного гильбертова пространства, в котором норма элемента есть квадратный корень из результата скалярного умножения элемента на себя. Наиболее известным примером такого пространства является пространство L_2 , в котором скалярное произведение элементов есть интеграл по области от произведения функций (элементов), взятый в некоторых случаях с заданным положительным весом ρ по области изменения аргумента.

При минимизации погрешности аппроксимации возникает следующая СЛАУ для коэффициентов элемента наилучшего приближения:

$$\sum_{j=0}^n \tilde{c}_j (\varphi_j, \varphi_i) = (f, \varphi_i).$$

Матрица данной системы представляет собой **матрицу Грама** используемой системы функций. В случае линейной независимости системы функций матрица Грама невырождена, поэтому элемент наилучшего приближения существует и является единственным.

Теорема 4.11. Если $\tilde{\varphi}$ — элемент наилучшего приближения в пространстве H , то погрешность $f - \tilde{\varphi}$ ортогональна $\tilde{\varphi}$, т. е.

$$(f - \tilde{\varphi}, \tilde{\varphi})_H = 0.$$

Следствие 4.2. Если $\tilde{\varphi}$ — элемент наилучшего приближения в пространстве H , то

$$\|f - \tilde{\varphi}\|_H^2 = \|f\|_H^2 - \|\tilde{\varphi}\|_H^2.$$

Доказательства теоремы 4.11 и следствия 4.2 опустим.

Теорема 4.11 показывает, что погрешность аппроксимации элементом наилучшего приближения ортогональна ему. Это означает, что погрешность лежит в подпространстве, ортогональном линейной оболочке, натянутой на рассматриваемую систему. Следствие 4.2 является аналогом теоремы Пифагора и позволяет вычислить норму погрешности.

Рассмотрим построение полинома наилучшего приближения с использованием ортонормированной системы функций $\{\varphi_i(x)\}$, $i = \overline{0, n}$. В этом случае коэффициенты обобщенного полинома могут быть найдены как

$$\tilde{c}_i = (f, \varphi_i), \quad i = \overline{0, n}.$$

Определение. Числа $\tilde{c}_i = (f, \varphi_i)$, $i = \overline{0, n}$, называются **коэффициентами Фурье** элемента $f \in H$ по ортонормированной системе $\{\varphi_i(x)\}$, $i = \overline{0, n}$, а обобщенный полином вида $\tilde{\varphi}(x) = \sum_{i=0}^n \tilde{c}_i \varphi_i(x)$ — **многочленом Фурье**.

Если система функций $\{\varphi_i(x)\}$, $i = 0, 1, \dots$, ортонормированная и полная, то имеет место равенство Парсеваля:

$$\|f\|_H^2 = \sum_{i=0}^{\infty} c_i^2.$$

Следовательно, ряд, составленный из квадратов коэффициентов Фурье, сходится, а его остаток стремится к нулю. В силу этого погрешность приближения многочленом Фурье стремится к нулю в смысле пространства L_2 . В результате аппроксимация возможна с любой наперед заданной точностью.

Успех аппроксимации обобщенным полиномом в конкретной ситуации определяется правильным выбором системы функций, с помощью которых выполняется такая аппроксимация, и соответствующего пространства, задаваемого, в частности, и способом вычисления в нем нормы. Использование неортогональной системы ведет к быстрому убыванию определителя решаемой системы и получению завышенных погрешностей аппроксимации. Поэтому лучше использовать ортогональную систему функций, нежели неортогональную, из которой она получена.

Например, матрица Грама естественной, казалось бы, системы функций $\{\varphi_i(x) = x^i\}$, $i = 0, 1, \dots$, дает стандартный пример плохо обусловленной матрицы (называемой *матрицей Гильберта*). Ее использование уже при $n > 5$ становится невозможным. Описанное свойство часто трактуется как переполненность степенного базиса.

Если аппроксимируемая функция известна только в точках сетки, то аппроксимация с использованием интегралов невозможна. В этом случае часто интеграл заменяют конечной суммой по точкам сетки, в которой перед каждым слагаемым стоит свой весовой коэффициент. Его значение, например, может отражать значимость данной конкретной точки или поведение функции.

Элемент наилучшего приближения далее находят с применением процедуры, практически аналогичной описанной выше. Однако значения коэффициентов элемента невозможно записать аналитически.

В случае использования конечномерного аналога пространства L_2 получается *метод наименьших квадратов*, часто применяемый при обработке экспериментальной информации.

Замечание 4.9. При рассмотрении экспериментальной информации, заданной приближенно, часто возникает задача аппроксимации с заданной точностью ε : среди всех обобщенных полиномов требуется найти полином, для которого $\|f - \varphi\|_H < \varepsilon$.

Очевидно, что для обеспечения единственности решения такой задачи необходимы дополнительные условия.

4.3.5. Насыщаемость алгоритма интерполяции.

Тригонометрическая интерполяция

Как уже было указано, при использовании равномерных сеток погрешность

$$|r_n| \leq \frac{M_{n+1}}{n+1} h^{n+1},$$

где $n + 1$ — число узлов сетки.

Улучшится ли качество интерполяции на данной сетке, если при фиксированном n рассмотреть функцию большей гладкости? Данная оценка показывает, что улучшения не произойдет. Погрешность и в этом случае останется величиной $O(h^{n+1})$.

Определение. *Алгоритм*, обладающий свойством независимости его погрешности от увеличения гладкости функции, называется **насыщаемым** (гладкостью).

Возникает вопрос: существуют ли **ненасыщаемые алгоритмы** интерполяции? Ответ на него положительный.

Примером ненасыщаемого алгоритма служит **тригонометрическая интерполяция** функциями вида

$$Q_n(x) = a_0 + \sum_{k=1}^n (a_k \cos k\omega x + b_k \sin k\omega x), \quad \omega = \frac{2\pi}{b-a}.$$

Для таких полиномов, совпадающих с периодической функцией (период $L = b - a$) в точках равномерной сетки $a = x_0 < x_1 < \dots < x_{2n} < b$ ($Q_n(x_i) = f(x_i)$, $i = \overline{0, 2n}$), имеют место следующие свойства:

1) погрешность

$$\|r_n\|_C = \|f - Q_n\|_C = O\left(\frac{M_{l+1}}{n^{l-1}}\right),$$

где M_{l+1} — оценка $(l+1)$ -й производной функции f , т. е. скорость убывания погрешности автоматически учитывает гладкость функции f ;

2) константа Лебега $\eta = O(\ln n)$, т. е. возрастает с увеличением n существенно медленнее, чем в обычной полиномиальной интерполяции на равномерной сетке.

Сопоставление условий, из которых определяются коэффициенты тригонометрического полинома, и его функционального вида позволяет записать полином в аналитическом виде (аналогично полиному Лагранжа, представленному выше):

$$Q_n(x) = \sum_{k=0}^{2n} f(x_k) q_k^n(x),$$

где $q_k^n(x)$ — **базисный тригонометрический полином**:

$$q_k^n(x) = \prod_{\substack{i=0 \\ i \neq k}}^{2n} \frac{\sin(0,5\omega(x - x_i))}{\sin(0,5\omega(x_k - x_i))}.$$

Можно показать, что система функций $1, \cos(k\omega x), \sin(k\omega x)$, $k = \overline{1, n}$, является базисом в пространстве сеточных функций, заданных на используемой интерполяционной сетке. При этом

она ортогональна относительно дискретного аналога обычного скалярного произведения в пространстве L_2 . Это позволяет записать аналитические выражения для коэффициентов $a_0, a_k, b_k, k = \overline{1, n}$, в виде дискретного аналога обычных выражений для коэффициентов Фурье.

Замечание 4.10. Анализ выражений для коэффициентов разложения $a_0, a_k, b_k, k = \overline{1, n}$, позволяет тривиально получить оценку константы Лебега вида $\eta = O(n)$. Однако более детальный анализ дает возможность установить на равномерной сетке логарифмическую оценку константы Лебега. Это, в частности, свидетельствует об оптимальности равномерной сетки для тригонометрической интерполяции. Расчетные формулы при этом получаются также очень простыми.

Замечание 4.11. Совпадение оценок констант Лебега тригонометрической интерполяции и полиномиальной на чебышёвской сетке неслучайно. Легко видеть, что замена x на φ по правилу

$$x = \frac{a+b}{2} + \frac{b-a}{2} \cos \varphi$$

приводит к преобразованию равномерной сетки переменной φ в чебышёвскую сетку переменной x . При этом тригонометрический полином переменной φ становится обычным полиномом переменной x .

4.4. Сплайн-интерполяция

Многих недостатков глобальной полиномиальной интерполяции лишена кусочно-многочленная, или кусочно-полиномиальная, интерполяция. Пусть имеются точки

$$a = x_0 < x_1 < \dots < x_n = b.$$

Найдем функцию $S_3(x)$, представляющую собой многочлен третьей степени на любом отрезке $[x_{i-1}, x_i]$ длиной $h_i = x_i - x_{i-1}$:

$$S_3(x) = a_i + b_i(x - x_{i-1}) + c_i(x - x_{i-1})^2 + d_i(x - x_{i-1})^3.$$

Необходимо на каждом из отрезков определить неизвестные коэффициенты a_i , b_i , c_i и d_i . Получим условия для их вычисления. Потребуем, чтобы на концах отрезка $[x_{i-1}, x_i]$, $i = \overline{1, n}$, функция S_3 принимала заданные значения $y_i = f(x_i)$:

$$S_3(x_{i-1}) = y_{i-1} = a_i;$$

$$S_3(x_i) = y_i = a_i + b_i h_i + c_i h_i^2 + d_i h_i^3.$$

Видно, что число неизвестных параметров пока в 2 раза превышает число полученных уравнений. Для увеличения числа уравнений потребуем также непрерывности первой и второй производных во внутренних точках сетки:

$$S'_3 = b_i + 2c_i(x - x_{i-1}) + 3d_i(x - x_{i-1})^2;$$

$$S''_3 = 2c_i + 6d_i(x - x_{i-1}).$$

Непрерывность производных означает, что

$$b_i + 2c_i h_i + 3d_i h_i^2 = b_{i+1};$$

$$2c_i + 6d_i h_i = 2c_{i+1}, \quad i = \overline{1, n-1}.$$

В результате получено $2n + 2(n-1) = 4n - 2$ уравнений для $4n$ неизвестных. Еще два уравнения можно записать, полагая производную S''_3 равной нулю в точках $x = x_0 = a$ и $x = x_n = b$:

$$2c_1 = 0; \quad 2c_n + 6d_n h_n = 0$$

(или полагая формально $c_{n+1} = 0$).

Для получения расчетных соотношений приведем рассматриваемую систему уравнений к удобному виду, исключив a_i, b_i, d_i :

$$d_n = -\frac{c_n}{3h_n}; \quad d_i = \frac{c_{i+1} - c_i}{3h_i};$$

$$b_i = \frac{y_i - y_{i-1}}{h_i} - c_i h_i - \frac{(c_{i+1} - c_i)h_i}{3}; \quad b_n = \frac{y_n - y_{n-1}}{h_n} - c_n h_n + \frac{c_n h_n}{3},$$

тогда равенство первых производных S'_3 дает

$$\begin{aligned} \frac{1}{h_i}(y_i - y_{i-1}) - c_i h_i - \frac{1}{3}(c_{i+1} - c_i)h_i + 2c_i h_i + (c_{i+1} - c_i)h_i = \\ = \frac{1}{h_{i+1}}(y_{i+1} - y_i) - c_{i+1} h_{i+1} - \frac{1}{3}(c_{i+2} - c_{i+1})h_{i+1}. \end{aligned}$$

Заменим индекс i на $i - 1$, чтобы уравнения приняли привычный вид:

$$\begin{aligned} c_{i-1}h_{i-1} + 2(h_{i-1} + h_i)c_i + c_{i+1}h_i = \\ = 3 \left[\frac{1}{h_i}(y_i - y_{i-1}) - \frac{1}{h_{i-1}}(y_{i-1} - y_{i-2}) \right]. \end{aligned}$$

При этом $c_1 = c_{n+1} = 0$. Равенство $c_{n+1} = 0$ легко получить из сравнения условий для S''_3 в точке $x_n = b$ и во внутренних точках.

Таким образом, имеем трехдиагональную СЛАУ со строгим диагональным преобладанием: разность коэффициента перед c_i и суммы коэффициентов перед c_{i+1} и c_{i-1} равна $h_{i-1} + h_i > 0$. Следовательно, задача определения c_i поставлена корректно, решение легко находится методом прогонки. Далее вычисляются остальные коэффициенты.

Построенная функция S_3 называется **интерполяционным кубическим сплайном** (от англ. spline — планка, рейка). Он также называется **естественным** или **чертежным сплайном**, поскольку происходит от следующего чертежного приема. Кривая строится по гибкой металлической линейке, которая проходит через заданные точки. Приложенная линейка принимает форму, соответствующую минимуму упругой энергии:

$$\int_a^b (u'')^2 dx \rightarrow \min.$$

Отсюда получается уравнение Эйлера $u^{(4)} = 0$. Его решение $u(x)$ есть многочлен третьей степени на каждом интервале сетки.

В узлах сетки должны быть непрерывны решение u и его производные u' и u'' .

Помимо относительной простоты сплайн-интерполяция замечательна еще и своей сходимостью. Приведем без доказательства следующую теорему.

Теорема 4.12. Пусть $u = f(x) \in C^4[a, b]$, $f''(a) = f''(b) = 0$, $M_4 = \|f^{(4)}\|_C$, $S_3(x)$ — сплайн третьей степени. Тогда

$$\|f - S_3\|_C \leq C_1 M_4 h^4; \|f' - S'_3\|_C \leq C_2 M_4 h^3; \|f'' - S''_3\|_C \leq C_3 M_4 h^2,$$

где C_1 , C_2 и C_3 — постоянные; h — шаг сетки.

Отсюда следует, что для указанного класса функций не только S_3 сходится к f , но и ее первая и вторая производные сходятся к соответствующим производным. Функцию S_3 можно дифференцировать.

Очевидно, что теорема 4.12 не имеет места для функций $f(x)$ таких, что $f''(a) \neq 0$ или $f''(b) \neq 0$. Например, для простейшего случая $n = 1$ и $f(x) = x^2$ на отрезке $[a, b] = [0; 1]$ имеем $S_3(x) = x$. Увеличение числа точек сетки не может дать норму погрешности $\|f'' - S''_3\|_C$, меньшую 2.

Замечание 4.12. Рассмотренная в 4.1.2 кусочно-линейная интерполяция — простейший пример не только интерполянта вообще, но и сплайна в частности.

Определение. **Интерполяционным сплайном степени m** называется функция $S_m(x)$, заданная на отрезке $[a, b]$ с указанными точками разбиения, непрерывная на этом отрезке вместе со своими производными вплоть до некоторого порядка p и совпадающая на каждом отрезке разбиения $[x_{i-1}, x_i]$, $i = \overline{1, n}$, с некоторым алгебраическим полиномом $P_{m,i}(x)$ степени m , такая что $S_m(x_i) = f_i$, $i = \overline{0, n}$. Разность $m - p$ между степенью сплайна и порядком наивысшей непрерывной производной называется **дефектом сплайна**.

Таким образом, кусочно-линейный интерполянт есть линейный (первой степени) сплайн с дефектом, равным единице. Дефект рассмотренного кубического сплайна также равен единице. Часто используются и кубические сплайны с дефектом, равным двум.

Очевидно, что для задания такого сплайна, вторая производная которого в точках сетки не является непрерывной, необходимо указать дополнительные условия. Такими условиями могут быть значения производной сплайна в точках разбиения $s_i = S'_m(x_i)$, называемые наклоном сплайна в точке x_i .

Нетрудно получить явное выражение для такого интерполянта (на отрезке разбиения $[x_{i-1}, x_i]$) без решения трехдиагональной СЛАУ:

$$\begin{aligned} S_3(x) = & \frac{(x - x_i)^2(2(x - x_{i-1}) + h_i)}{h_i^3} y_{i-1} + \\ & + \frac{(x - x_{i-1})^2(2(x_i - x) + h_i)}{h_i^3} y_i + \frac{(x - x_i)^2(x - x_{i-1})}{h_i^2} s_{i-1} + \\ & + \frac{(x - x_{i-1})^2(x - x_i)}{h_i^2} s_i. \end{aligned}$$

Существуют различные способы задания наклонов сплайна в точках сетки. Наиболее простой способ задания s_i получается в случае, если известны производные исходной функции в точках x_i , $i = \overline{0, n}$. Тогда $s_i = f'(x)|_{x=x_i}$. Такой сплайн называется локальным, так как на каждом отрезке разбиения $[x_{i-1}, x_i]$ он полностью определяется значениями функции и ее производной на границах отрезка. Очевидно, что этот сплайн представляет собой интерполяционный полином Эрмита на каждом таком отрезке.

Построенный выше кубический сплайн с дефектом, равным единице, естественно считать глобальным сплайном, так как его коэффициенты определяются данными на всей сетке сразу.

Замечание 4.13. Довольно часто в рассмотрение вводятся так называемые **B-сплайны**. Они представляют собой кусочно-

полиномиальные функции, задаваемые полиномом в областях его неотрицательности и нулем в остальной части числовой оси (в одномерном случае). Так, B -сплайн нулевой степени есть характеристическая функция одного полуинтервала между точками сетки.

Линейный B -сплайн ранее рассматривался при описании базисных функций линейного конечного элемента (см. 4.1.2). Такие сплайны широко используют при решении задач математической физики методом конечных элементов.

4.5. Двумерная интерполяция

Сетка на плоскости переменных x, y представляет собой набор точек

$$\Omega_h = \{(x_i, y_i), a \leq x_i \leq b, c \leq y_i \leq d, i = \overline{1, N}\},$$

связанных друг с другом и образующих ячейки некоторой выбранной формы. В зависимости от формы ячеек выделяют сетки прямоугольные, треугольные, четырехугольные, смешанные и др. Различают также структурированные и неструктурные сетки в зависимости от того, насколько упорядочено положение ячеек в сетке.

Пусть в каждой точке Ω_h задано значение функции z_i , $i = \overline{1, N}$. Необходимо проинтерполировать эти значения, т. е. построить двумерную функцию $\tilde{f}(x, y)$ вместо $z = f(x, y)$. При этом $\tilde{f}(x, y)$ должна «приближать» функцию $f(x, y)$.

Рассмотрим два типа сеток — прямоугольную и треугольную.

Прямоугольная сетка. В случае прямоугольной сетки

$$\Omega_h = \Omega_h^x \times \Omega_h^y,$$

где

$$\Omega_h^x = \{x_i : a \leq x_i \leq b, i = \overline{0, n}\};$$

$$\Omega_h^y = \{y_j : c \leq y_j \leq d, j = \overline{0, m}\}.$$

Введем обозначение: $z_{ij} = f(x_i, y_j)$.

Наиболее простой вариант построения интерполяционного многочлена заключается в отдельной интерполяции по x и y и выборе результирующего интерполянта в виде произведения одномерных интерполянтов. Такая **интерполяция** называется **последовательной**. Считаем, что $a = x_0$, $b = x_n$, $c = y_0$, $d = y_m$.

Пусть для произвольных $y = y_j$ справедливо равенство

$$\tilde{f}(x, y_j) = \sum_{i=0}^n z_{ij} \varphi_i^x(x),$$

а для произвольного $x = x_i$

$$\tilde{f}(x_i, y) = \sum_{j=0}^m z_{ij} \varphi_j^y(y).$$

Тогда

$$\tilde{f}(x, y) = \sum_{i=0}^n \sum_{j=0}^m z_{ij} \varphi_i^x(x) \varphi_j^y(y),$$

т. е. получена последовательная интерполяция.

В качестве φ_i^x , φ_j^y можно взять функции типа базисных функций конечных элементов, полинома Лагранжа вида $\frac{\omega(x)}{(x - x_i)\omega'(x_i)}$ (для x), а также функции типа сплайна. Базисные сплайны соответствуют единичному значению функции в i -й точке и нулю в остальных точках.

Однако при последовательной интерполяции завышается степень интерполяционного многочлена. Так, если φ_i^x , φ_j^y — линейные одномерные базисные функции конечных элементов, то на сеточных прямоугольниках интерполянт имеет вторую степень:

$$\varphi_i^x \varphi_j^y = a + bx + cy + dxy,$$

но при этом никакого повышения точности интерполяирования по сравнению с точностью линейной одномерной (по x или y) интерполяции не происходит.

Рассмотрим многочлен N -й степени двух переменных:

$$z = f(x, y) = \sum_{\substack{i+j=0 \\ 0 \leq i, j \leq N}}^N a_{ij} x^i y^j.$$

В этом выражении использовано следующее количество коэффициентов:

$$\begin{aligned} \frac{1}{2} \left[(N+1)^2 - (N+1) \right] + (N+1) &= \\ &= \frac{1}{2}(N+1)^2 + \frac{1}{2}(N+1) = \frac{1}{2}(N+1)(N+2). \end{aligned}$$

Соответственно, для их определения необходимо задать столько же уравнений. Если есть $n+1$ точек по x и $m+1$ точек по y , то должно быть выполнено равенство

$$(n+1)(m+1) = \frac{1}{2}(N+1)(N+2).$$

Удовлетворить этому условию довольно сложно. Например, если $n = m = 1$, то должно быть выполнено равенство

$$(1+1)(1+1) = 4 = \frac{1}{2}(N+1)(N+2),$$

откуда

$$N^2 + 3N + 2 = 8;$$

$$N = -\frac{3}{2} \pm \sqrt{\frac{9}{4} + 6}.$$

Выберем положительное значение $N = \frac{\sqrt{33}-3}{2}$, однако отметим, что оно не является целым числом.

При $N = 1$

$$\frac{1}{2}(N+1)(N+2) = 3,$$

при $N = 2$

$$\frac{1}{2}(N+1)(N+2) = 6.$$

В результате для полинома первой степени с $N = 1$ четырех точек много, а для полинома второй степени с $N = 2$ — мало. Если выбрать $N = 2$, то необходимо добавить еще два уравнения. Их выбор неоднозначен, что создает проблемы при построении двумерных интерполянтов.

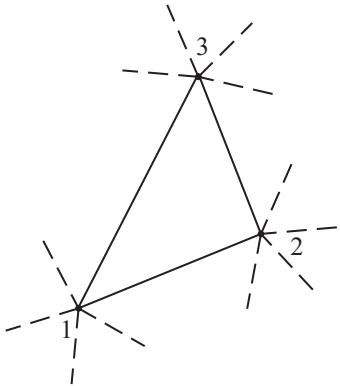


Рис. 4.10. Ячейка треугольной сетки

Треугольная сетка. Для построения полинома первой степени с $N = 1$ требуется лишь три точки сетки (рис. 4.10). Таким образом, полином минимальной степени получается на треугольной сетке. Легко записать функцию $\tilde{f}(x, y) = a + bx + cy$, которая принимает в вершинах треугольника заданные значения:

$$\begin{aligned} a + bx_1 + cy_1 &= f_1; \\ a + bx_2 + cy_2 &= f_2; \\ a + bx_3 + cy_3 &= f_3. \end{aligned}$$

Эта СЛАУ относительно коэффициентов a , b , c имеет однозначное решение, если точки 1, 2, 3 (см. рис. 4.10) не лежат на одной прямой.

Если рассматривать интерполяцию на всей сетке, то получим

$$z = \tilde{f}(x, y) = \sum_{k=1}^K z_k \varphi_k^{xy},$$

где K — число узлов сетки; φ_k^{xy} — кусочно-линейная двумерная конечно-элементная базисная функция.

График линейной базисной функции двумерного конечного элемента представляет собой пирамиду (рис. 4.11). Функция φ_k^{xy} принимает значение 1 в k -й точке и 0 — во всех остальных вершинах треугольников, имеющих k -ю точку своей вершиной.

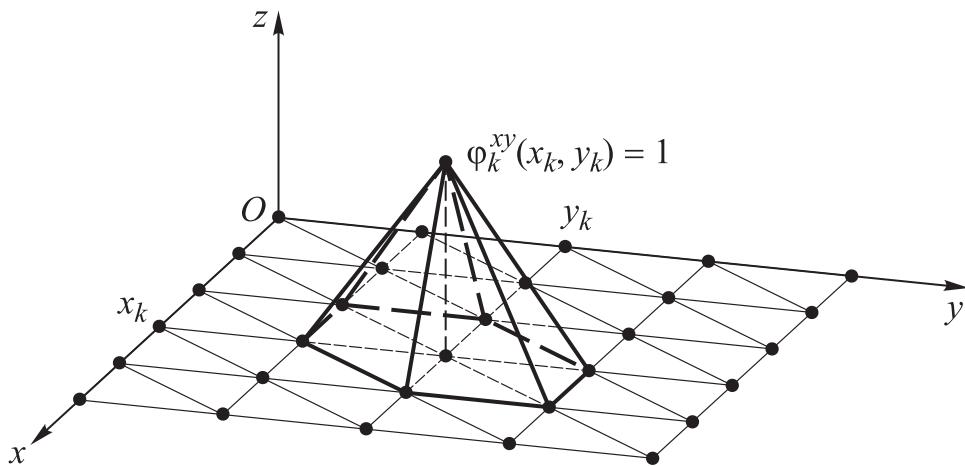


Рис. 4.11. График линейной базисной функции на двумерной конечно-элементной сетке

На каждом треугольнике функция φ_k^{xy} представляет собой плоскость. Носитель функции φ_k^{xy} , т. е. множество, на котором данная функция отлична от нуля, называется **двумерным конечным элементом**.

Для построения полинома второго порядка ($N = 2$) требуется шесть точек. Чаще всего это три угловые точки (см. рис. 4.10) и три точки в центрах сторон треугольника. Для полинома третьего порядка ($N = 3$) требуется десять точек. Обычно это три угловые точки, по две точки на сторонах треугольника и одна точка в его центре.

Существует развитая теория конечных элементов, а также различные технологии работы с ними.

Отметим очевидное отличие **двумерной экстраполяции** от **одномерной** (рис. 4.12): вычисление \tilde{f} в точке B , которая находится вне выпуклого тела, есть экстраполяция, а в точке A , которая находится внутри него, — интерполяция.

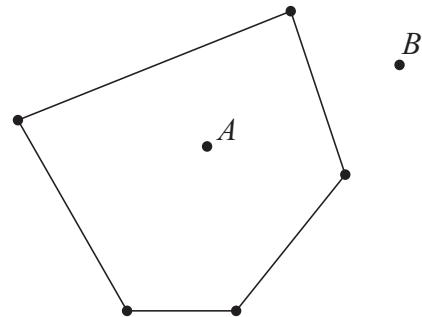


Рис. 4.12. Интерполяция и экстраполяция в двумерном случае

Вопросы и задания

1. Сформулируйте задачи интерполяции, экстраполяции и приближения функции. В чем сходства и различия этих задач?
2. Сформулируйте алгоритм линейной интерполяции. Как можно получить оценку погрешности линейной интерполяции?
3. Дайте определение линейного одномерного конечного элемента.
4. Какая интерполяция называется полиномиальной? Какие интерполяционные полиномы вы знаете?
5. Запишите остаточный член полиномиальной интерполяции. Оцените точность приближения функции полиномом Лагранжа.
6. Запишите остаточный член полиномиальной экстраполяции.
7. Что такое разделенные разности? Приведите примеры их представления.
8. Приведите две формы записи интерполяционного полинома — Лагранжа и Ньютона. В чем достоинства и недостатки каждой формы записи интерполяционного полинома?
9. Какой интерполяционный полином называется полиномом Эрмита? Каковы его достоинства и недостатки?
10. Какой полином называется полиномом Чебышёва? Каковы свойства этого полинома? Какое практическое применение имеют полиномы Чебышёва?
11. Как зависит погрешность полиномиальной интерполяции от типа используемых сеток?

12. Что такое константа Лебега? Чему равна константа Лебега для равномерной сетки, для чебышёвской сетки?
13. Как зависит погрешность полиномиальной интерполяции от погрешности задания функции?
14. Какой полином называется полиномом наилучшего приближения функции и каковы его характеристики?
15. Приведите оценку погрешности полиномиальной аппроксимации на чебышёвской сетке.
16. Что называется насыщаемостью алгоритма интерполирования? Какие алгоритмы интерполирования насыщаемы? Приведите пример ненасыщаемого алгоритма.
17. Какая интерполяция называется тригонометрической? Какие свойства тригонометрической интерполяции вы знаете?
18. Что называется сплайном? Какая интерполяция называется сплайн-интерполяцией? Дайте оценку погрешности интерполирования кубическими сплайнами.
19. Приведите алгоритмы двумерной интерполяции и экстраполяции.

Библиографические комментарии

Теория аппроксимации и ее частного случая — интерполяции относится к наиболее разработанным областям математики, используемым в приложениях. Эта теория опирается на функциональный анализ, теорию функций, численный анализ. Для изучения теории аппроксимации можно обратиться к работам [6, 53], в которых содержится материал по аппроксимации классов функций на конечномерных компактах. Там же имеется большое количество ссылок на соответствующую литературу.

Одна из наиболее интересных задач теории аппроксимации — задача построения равномерных приближений. Многие результаты решения этой задачи можно найти в работах [10] и [11]. Вопросы их практического применения отражены в [25, 35, 37].

Теория сплайнов и их применению в вычислительной математике посвящена, например, монография [72]. Сплайн-интерполяция, теория и вопросы ее практического применения (включая B -сплайны) довольно подробно описаны в работе [15], а также в [9, 35, 40, 47, 65].

Отметим, что приближение функций с помощью конечных сумм — широко распространенный прием получения чисто математических результатов типа существования и единственности решения краевых задач, вариационных неравенств и т. п. Такой прием использован в работах [13, 48, 51, 52, 64, 73].

Материал по специальным функциям математической физики можно найти, например, в [16, 50, 76]. Очень много информации по специальным функциям содержит справочник [1]. Особо отметим также монографию [58].

Алгоритмы без насыщения представляют собой один из важнейших классов численных алгоритмов. Их построение довольно сложно. Значительная информация о таких алгоритмах содержится в работах [6] и [53].

5. РЕШЕНИЕ НЕЛИНЕЙНЫХ УРАВНЕНИЙ

Представлены наиболее распространенные методы решения скалярного нелинейного уравнения и систем нелинейных уравнений. Обоснованы метод деления отрезка пополам, методы хорд и Ньютона. Доказана сходимость стационарных методов, сводимых к поиску неподвижной точки отображения. Рассмотрены внутренние и внешние итерации, применяемые для нахождения решения систем нелинейных уравнений. Приведены примеры гибридных методов для решения таких систем.

5.1. Решение скалярных уравнений

5.1.1. Постановка задачи и основные процедуры решения

Рассмотрим задачу поиска корней скалярного уравнения $f(x) = 0$, где $f(x)$ — некоторая заданная функция. Известно, что корни уравнения $f(x) = 0$ могут быть действительными и комплексными, простыми и кратными. Поэтому выделяют несколько задач, связанных с их поиском:

- 1) определение области локализации корней;
- 2) нахождение корня;
- 3) определение кратности корня;
- 4) уточнение значений найденных корней и оценка их точности.

Ограничимся рассмотрением лишь одной задачи: поиск простого (некратного) корня уравнения $f(x) = 0$, расположенного на отрезке $[a, b]$ непрерывности функции $f(x)$, причем $f(a)f(b) < 0$, для определенности $f(a) < 0, f(b) > 0$ (если это не так, то всегда

вместо $f(x)$ можно взять $-f(x)$, что ничего не меняет). При этих условиях, как известно из курса математического анализа (теорема Коши о значениях непрерывной на отрезке функции), на отрезке $[a, b]$ существует корень уравнения $f(x) = 0$.

Для корректного решения поставленной задачи нужно быть уверенным в наличии не более одного корня на рассматриваемом отрезке $[a, b]$. Разные знаки функции на границах отрезка гарантируют только нечетность числа простых действительных корней на данном отрезке. Для того чтобы убедиться, единственный ли корень на отрезке, в одномерном случае можно использовать аппарат классического математического анализа, применяемый при качественном исследовании (нахождение областей монотонности, выпуклости и т. п.) графика функции. С помощью этого аппарата можно определить также области локализации корней.

Замечание 5.1. Задачу о локализации корней можно решить аналитически или графически. При поиске действительных корней удобно составлять таблицу значений $x_i, f(x_i)$. Если в двух соседних строках таблицы значения функции $f(x)$ имеют разные знаки, то между соответствующими x_i и x_{i+1} есть по меньшей мере один корень. Однако с помощью таблицы трудно определить корень с четной кратностью. Построение графика $f(x)$ может помочь в решении задачи локализации корня и поиска корней с четной кратностью. Табулирование функции является первым шагом использования интерполяции, а точнее — обратной интерполяции, поскольку корень уравнения может быть найден как $x = f^{-1}(0)$.

Замечание 5.2. Поиск полного набора корней уравнения ведется последовательно, один корень за другим. При этом после нахождения простого корня x_n необходимо исходную функцию разделить на $x - x_n$. Далее нужно искать решение модифицированной задачи. В случае кратного корня потребуется соответствующая модификация алгоритма.

Отметим, что значения корней по-разному зависят от погрешности задания исходной функции. Известно, что наибольшие по абсолютному значению корни наименее устойчивы. Поэтому поиск всех корней необходимо начинать с меньших корней, после чего делить функцию на $x - x_n$ и продолжать процесс.

Нелинейные уравнения решаются, как правило, итерационными методами, т. е. строится последовательность приближений $\{x^k\}_{k=0}^{\infty}$ к искомому корню x_* , где k — номер итерации. Величина $z^k = x^k - x_*$ называется **погрешностью приближения** x^k .

Итерационный метод решения скалярного нелинейного уравнения называется **сходящимся**, если $|z^k| \rightarrow 0$ при $k \rightarrow \infty$. Обычно требуется найти приближение к корню с погрешностью не больше $\varepsilon > 0$, т. е. так, чтобы

$$|x^k - x_*| \leq \varepsilon \text{ при } k \geq k_0(\varepsilon).$$

Если это условие выполнено, то вычисления можно прекратить при $k = k_0$. На практике точное решение неизвестно и использовать неравенство $|x^k - x_*| \leq \varepsilon$ в качестве критерия прекращения итераций не представляется возможным. Поэтому чаще всего применяют следующие условия для остановки итерационного процесса:

$$|f(x^{k+1})| \leq \varepsilon$$

или

$$|x^{k+1} - x^k| \leq \varepsilon.$$

Однако последнее условие может приводить к неверному заключению о сходимости метода, если, например, метод сходится очень медленно. Поэтому при решении нелинейных уравнений основными вопросами являются сходимость итерационного процесса и скорость сходимости, т. е. минимальное количество итераций, при котором достигается требуемая точность.

5.1.2. Метод «вилки», или деления отрезка пополам

Одним из наиболее простых, но в то же время надежных методов решения скалярных нелинейных уравнений является **метод «вилки»**, называемый также **методом бисекции** или **методом деления отрезка пополам**. Пусть на отрезке $[a, b]$ существует один корень уравнения $f(x) = 0$, $f(a)f(b) < 0$, для определенности $f(a) < 0$, $f(b) > 0$. Зададим $a_0 = a$, $b_0 = b$ и найдем середину отрезка $[a_0, b_0]$:

$$c_1 = (a_0 + b_0)/2.$$

Вычислим $f(c_1)$. Если $f(c_1) = 0$, то решение задачи найдено. Если же $f(c_1) > 0$, то $a_1 = a_0$, $b_1 = c_1$, в противном случае $a_1 = c_1$, $b_1 = b$. Поиск корня продолжается на отрезке $[a_1, b_1]$ таком, что $f(a_1)f(b_1) < 0$. Далее процедура повторяется, на каждом шаге отрезок локализации корня уменьшается вдвое. В результате на n -м шаге решение уравнения ищется на отрезке длиной

$$b_n - a_n = (b - a)/2^n, \quad n = 0, 1, \dots.$$

Если в качестве приближения к решению x_* взять среднее $x = (a_n + b_n)/2$, то

$$|x_* - x| \leq \frac{1}{2}(b - a)2^{-n}.$$

Тогда для нахождения решения с погрешностью не больше $\varepsilon > 0$ требуется выполнить следующее количество итераций:

$$n_0 = \left[\ln \left(\frac{b - a}{2\varepsilon} \right) / \ln 2 \right] + 1,$$

где $[\cdot]$ — целая часть числа.

В принятых нами условиях (наличие на отрезке $[a, b]$ одного корня нечетной кратности) метод «вилки» не может не сойтись. Однако для этого потребуется $n_0 N_f$ действий, где N_f — количество операций, необходимых для вычисления функции $f(x)$. Оно может оказаться непозволительно большим,

если решать скалярное уравнение требуется много раз подряд. Существуют другие, более быстрые методы, например метод хорд или метод Ньютона.

5.1.3. Итерационные методы типа простой итерации

Запишем исходное уравнение $f(x) = 0$ в виде $x = F(x)$; функцию $F(x)$ можно задать как $F(x) = x + \tau(x)f(x)$, где $\tau(x)$ — знакопостоянная на отрезке $[a, b]$ функция. **Метод типа простой итерации** определяется формулой

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

где k — номер итерации; x^0 — заданное начальное приближение.

Теорема 5.1 (сходимость методов типа простой итерации). Пусть функция $F(x)$ липшиц-непрерывна с постоянной $q \in (0; 1)$ на отрезке $[c - \delta, c + \delta] = \tilde{O}_\delta(c)$, т. е. для любых $x', x'' \in \tilde{O}_\delta(c)$ справедливо неравенство

$$|F(x') - F(x'')| \leq q|x' - x''|,$$

причем $|F(c) - c| \leq (1 - q)\delta$. Тогда уравнение $x = F(x)$ имеет единственное решение x_* на отрезке $\tilde{O}_\delta(c)$, которое можно найти в результате описанного выше итерационного процесса при любом $x^0 \in \tilde{O}_\delta(c)$. Для погрешности справедлива оценка

$$|x^k - x_*| \leq q^k |x^0 - x_*|, \quad k = 0, 1, \dots,$$

или

$$|x^k - x_*| \leq \frac{q^k}{1 - q} |F(x^0) - x^0|, \quad k = 0, 1, \dots.$$

◀ Пусть $x^0 \in \tilde{O}_\delta(c)$. Допустим, что и $x^k \in \tilde{O}_\delta(c)$. Докажем, что $x^{k+1} \in \tilde{O}_\delta(c)$. Действительно,

$$x^{k+1} - c = F(x^k) - F(c) + F(c) - c,$$

откуда

$$|x^{k+1} - c| \leq q|x^k - c| + (1-q)\delta \leq q\delta + (1-q)\delta = \delta.$$

Следовательно, $x^{k+1} \in \tilde{O}_\delta(c)$, т. е. итерационный процесс устроен таким образом, что последующие приближения x^{k+1}, x^{k+2}, \dots не выходят за пределы отрезка локализации $\tilde{O}_\delta(c)$.

Оценим разность двух соседних приближений $x^{k+1} - x^k$. Поскольку $x^{k+1} - x^k = F(x^k) - F(x^{k-1})$, имеем

$$\begin{aligned} |x^{k+1} - x^k| &\leq q|x^k - x^{k-1}| \leq q^k|x^1 - x^0| = q^k|F(x^0) - x^0|, \\ k &= 1, 2, \dots. \end{aligned}$$

Тогда

$$\begin{aligned} x^{k+p} - x^k &= \sum_{l=1}^p (x^{k+l} - x^{k+l-1}); \\ |x^{k+p} - x^k| &\leq |F(x^0) - x^0| \sum_{l=1}^p q^{k+l-1} = \\ &= |F(x^0) - x^0| q^k \frac{1 - q^p}{1 - q} \leq \frac{q^k}{1 - q} |F(x^0) - x^0|. \end{aligned}$$

Отсюда заключаем, что $\{x^k\}$ — фундаментальная последовательность. Она имеет предел, находящийся на отрезке $\tilde{O}_\delta(c)$. Поскольку $F(x)$ — непрерывная функция, то, переходя к пределу в соотношении $x^{k+1} = F(x^k)$ при $k \rightarrow \infty$, получим

$$\lim_{k \rightarrow \infty} x^{k+1} = \lim_{k \rightarrow \infty} x^k = F\left(\lim_{k \rightarrow \infty} x^k\right),$$

откуда

$$x_* = \lim_{k \rightarrow \infty} x^k; \quad x_* \in \tilde{O}_\delta(c).$$

Устремив p к бесконечности ($p \rightarrow \infty$) в неравенстве для $|x^{k+p} - x^k|$, получим оценку погрешности решения через известные величины:

$$|x^k - x_*| \leq \frac{q^k}{1 - q} |F(x^0) - x^0|.$$

Поскольку $x^{k+1} = F(x^k)$, $x_* = F(x_*)$, то

$$|x^{k+1} - x_*| = |F(x^k) - F(x_*)| \leq q|x^k - x_*| \leq q^{k+1}|x^0 - x_*|,$$

или

$$|x^k - x_*| \leq q^k|x^0 - x_*|.$$

Данный корень — единственный на отрезке $\tilde{O}_\delta(c)$, так как, предположив наличие двух корней x_*^1 и x_*^2 , получим

$$|x_*^1 - x_*^2| = |F(x_*^1) - F(x_*^2)| \leq q|x_*^1 - x_*^2|,$$

откуда $x_*^1 = x_*^2$, поскольку $q \in (0; 1)$. ►

Следствие 5.1. Если вместо условия липшиц-непрерывности функции $F(x)$ верно неравенство $|F'(x)| \leq q < 1$ на отрезке $\tilde{O}_\delta(c)$, то все условия теоремы 5.1 выполнены и ее выводы справедливы.

Следствие 5.2. Пусть функция $F(x)$ непрерывно дифференцируема на отрезке $\tilde{O}_\varepsilon(x_*)$ ($x_* = F(x_*)$ — решение) и $|F'(x_*)| < 1$. Тогда существует такое $\delta > 0$, что в δ -окрестности $\tilde{O}_\delta(x_*) \subset O_\varepsilon(x_*)$ уравнение $x = F(x)$ имеет единственный корень и итерационный процесс на отрезке $\tilde{O}_\delta(x_*)$ сходится, если $x^0 \in \tilde{O}_\delta(x_*)$.

◀ В силу непрерывности $F'(x)$ на отрезке $O_\varepsilon(x_*)$ существует δ -окрестность $\tilde{O}_\delta(x_*)$ точки x_* , в которой $|F'(x)| \leq q^* < 1$. Тогда выполнены все условия теоремы: $c = x_*$; $|F(c) - c| = 0 < (1 - q^*)\delta$. Поэтому утверждение следствия справедливо. ►

Если погрешность метода удовлетворяет оценке

$$|x^k - x_*| \leq Lq^k|x^0 - x_*|,$$

то говорят, что имеет место **линейная сходимость метода со скоростью геометрической прогрессии** с показателем q . В условиях теоремы 5.1 коэффициент $L = 1$, на каждой итерации выполнено неравенство $|x^k - x_*| \leq q|x^{k-1} - x_*|$.

Замечание 5.3. Сходимости методов типа простой итерации можно дать следующую геометрическую интерпретацию: поиск

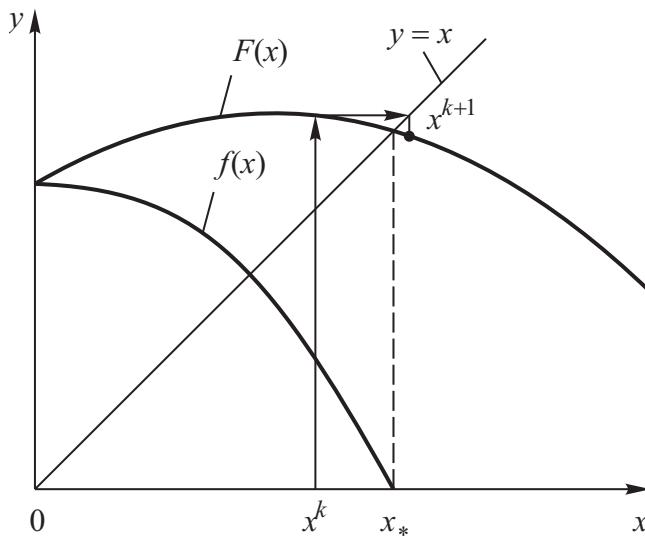


Рис. 5.1. Геометрическая интерпретация сходимости метода простой итерации

корня уравнения $f(x) = 0$ заменяют поиском абсциссы точки пересечения прямой $y = x$ и графика $y = F(x)$. Каждое следующее приближение к корню является абсциссой точки пересечения прямых $y = x$ и $y = F(x^k)$. На рис. 5.1 показан сходящийся итерационный процесс. Для этого в качестве начального приближения выбран x^0 из отрезка, где $|F'| < 1$.

Частным случаем метода простой итерации является **нелинейный метод релаксации**, для которого функция

$$F(x) = x + \tau f(x).$$

Тогда итерационный процесс имеет вид

$$\frac{x^{k+1} - x^k}{\tau} = f(x^k)$$

и $F'_x = 1 + \tau f'_x$. По теореме 5.1 метод сходится при $|1 + \tau f'_x| < 1$, т. е. при $-2 < \tau f'(x) < 0$. Если в некоторой окрестности корня производная функции $f(x)$ отрицательна и ограничена по модулю значениями m и M : $f'_x < 0$, $0 < m < |f'_x| < M$, то метод сходится при $\tau < 2/M$.

Пример 5.1. Рассмотрим процесс поиска корней нелинейного уравнения

$$x^3 - 20x + 1 = 0.$$

Для локализации корней исследуем функцию $f(x) = x^3 - 20x + 1$ и построим ее график (рис. 5.2).

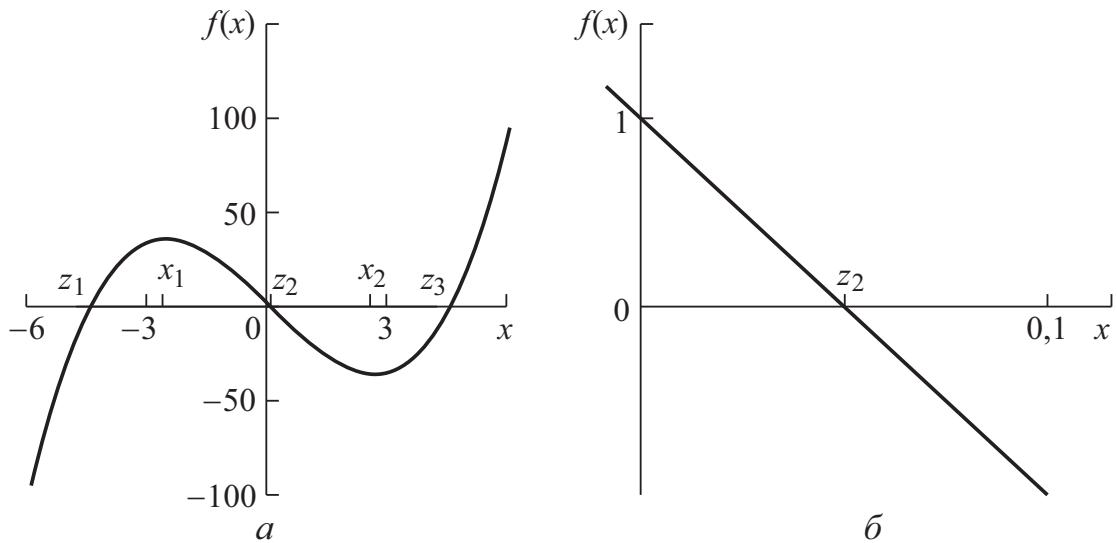


Рис. 5.2. Общий вид графика функции $f(x)$ из примера 5.1 (а) и фрагмент графика в окрестности среднего корня (б)

Очевидно, что $\lim_{x \rightarrow -\infty} f(x) = -\infty$ и $\lim_{x \rightarrow +\infty} f(x) = +\infty$, а $f(0) = 1$.

Точки экстремума можно найти из условия

$$f'(x) = 3x^2 - 20 = 0.$$

Это точки $x_1 = -\sqrt{20/3}$ и $x_2 = \sqrt{20/3}$. Несложно проверить, что $f(x_1) > 0$, а $f(x_2) < 0$. Таким образом, функция $f(x) = x^3 - 20x + 1$ имеет три корня на интервалах $(-\infty, -\sqrt{20/3})$, $(0, \sqrt{20/3})$ и $(\sqrt{20/3}, +\infty)$. Подобная локализация очень грубая, но для нашего исследования окажется достаточной.

Итерационный процесс типа простой итерации запишем в виде

$$x^{k+1} = \frac{1}{20} \left[(x^k)^3 + 1 \right],$$

причем $\tau = \frac{1}{20}$. Разность двух соседних приближений

$$x^{k+1} - x^k = \frac{1}{20} \left[(x^k)^3 - 20x^k + 1 \right],$$

и ее знак совпадает со знаком $f(x)$.

Из графика функции $f(x)$, приведенного на рис. 5.2, видно, что при выборе начального приближения $x^0 < z_1$ и $x^0 > z_3$ итерационный процесс оказывается расходящимся: каждое следующее приближение сдвигается по оси влево (при $x^0 < z_1$) или вправо (при $x^0 > z_3$) и удаляется от корня.

При выборе $x^0 \in [z_1, z_3]$ процесс может сходиться, но только к корню z_2 . На отрезке $[x_1, x_2]$ производная $f'(x) = 3x^2 - 20 < 0$ и $|f'(x)| < 20$, следовательно, условие $\tau < 2/M$ выполняется и итерационный процесс сходится. •

Найдем оптимальное значение параметра τ , т. е. значение, при котором метод сходится за наименьшее количество итераций. Пусть $z^k = x^k - x_*$ — погрешность приближения x^k к корню. Тогда

$$\begin{aligned} \frac{z^{k+1} - z^k}{\tau} &= f(x_* + z^k) = f(x_* + z^k) - f(x_*) = f'(x_* + \theta z^k) z^k; \\ z^{k+1} &= z^k \left[1 + \tau f'(x_* + \theta z^k) \right], \quad \theta \in (0; 1). \end{aligned}$$

Отсюда оценка погрешности z^{k+1}

$$|z^{k+1}| \leq \max_y |1 + \tau f'(y)| |z^k| \leq \left(\max_y |1 + \tau f'(y)| \right)^{k+1} |z^0|.$$

Однако

$$\max_y |1 + \tau f'(y)| = \max \{|1 - \tau m|, |1 - \tau M|\}.$$

Выберем τ , которое минимизирует эту величину (определение оптимального τ проиллюстрировано на рис. 5.3):

$$\tau = 2/(m + M).$$

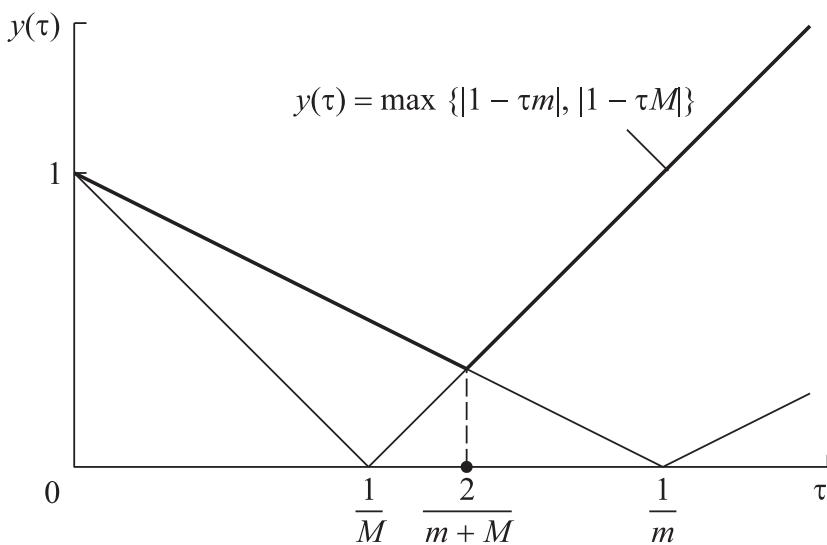


Рис. 5.3. Определение оптимального итерационного параметра τ

Отсюда получим

$$|z^k| \leq \left(\frac{M-m}{M+m}\right)^k |z^0|,$$

или

$$|z^k| \leq \left(\frac{1-\xi}{1+\xi}\right)^k |z^0|, \quad \xi = \frac{m}{M}.$$

Аналогичная задача о минимаксе была рассмотрена при исследовании итерационных методов вариационного типа (см. 3.2) и при решении задачи построения наилучшего равномерного приближения функции (см. 4.3.3).

Пример 5.2. Решим методом релаксации уравнение

$$1 - x^2 = 0,$$

т. е. $f(x) = 1 - x^2$. Несложно проверить, что на отрезке $[0,1; 2]$ существует корень $x_* = 1$, при этом в рассматриваемой области $f'(x) < 0$ и $m = 0,2 < |f'(x)| < 4 = M$. Таким образом, метод релаксации будет сходиться к корню $x_* = 1$, если $\tau < 2/M = 0,5$.

Пусть $\tau = 0,1$, начальное приближение $x^0 = 2$. Организуем итерационный процесс:

$$x^1 = x^0 + \tau f(x^0) = 2 + 0,1(1 - 2^2) = 1,7;$$

$$x^2 = x^1 + \tau f(x^1) = 1,7 + 0,1(1 - 1,7^2) = 1,511 \text{ и т. д.}$$

На 29-й итерации погрешность не превышает $\varepsilon = 10^{-3}$, $x^{29} \approx 1,0009293$, при этом $x^{28} \approx 1,0011617$.

Если изменить значение итерационного параметра τ , то скорость сходимости может быть выше. В табл. 5.1 приведено количество итераций, необходимое для достижения погрешности не более $\varepsilon = 10^{-3}$ при различных начальных приближениях и значениях τ .

Таблица 5.1

Количество итераций для решения уравнения $f(x) = 1 - x^2$ методом релаксации с погрешностью не более $\varepsilon = 10^{-3}$

τ	Количество итераций при значениях x^0				
	0,1	0,5	1,5	1,75	2
0,1	34	30	27	28	29
0,25	12	10	9	9	9
2/4,2	4	3	3	3	4
0,5	4	3	3	3	4

В результате получили, что наибольшая скорость сходимости достигается при τ , близких к 0,5, при любом начальном приближении из отрезка $[0,1; 2]$. •

5.1.4. Интерполяционные методы

В основе интерполяционных методов лежат следующие действия: замена функции $f(x)$ ее приближением $\tilde{f}(x)$, сконструированным путем *интерполяции* (иногда *экстраполяции*) по нескольким точкам, и выбор в качестве приближенного корня

уравнения $f(x) = 0$ корня уравнения $\tilde{f}(x) = 0$. Большинство наиболее часто применяемых методов этого класса можно представить в виде метода типа простой итерации, поэтому будем использовать теорему 5.1 для доказательства их сходимости.

Метод хорд (секущих). Построим метод на основе линейной интерполяции $\tilde{f}(x)$. Замена функции $f(x)$ прямой, проходящей через две точки x^k, x^{k+1} , дает

$$\tilde{f}(x) = f(x^k) + \left[f(x^{k+1}) - f(x^k) \right] \frac{x - x^k}{x^{k+1} - x^k}.$$

Тогда получаем **метод секущих (хорд):**

$$x^{k+2} = x^k + f(x^k) \frac{x^{k+1} - x^k}{f(x^k) - f(x^{k+1})}.$$

Для организации такого итерационного процесса потребуются два начальных приближения: x^0 и x^1 . Для того чтобы этого избежать, как правило, применяют более простую модификацию указанного метода, предусматривающую, что вместо x^k на каждой итерации берется некоторая фиксированная точка. В качестве такой точки удобно использовать одну из границ отрезка локализации корня.

Рассмотрим последний вариант подробнее, заменив в формулах метода x^k на b , и запишем метод хорд в виде метода типа простой итерации. В этом случае

$$F(x) = x - f(x) \frac{b - x}{f(b) - f(x)}.$$

Полагаем, что при $x = b$

$$F(b) = b - f(b)/f'(b).$$

Такой выбор $F(x)$ соответствует следующему итерационному процессу: на каждой итерации строится секущая (хорда) к графику

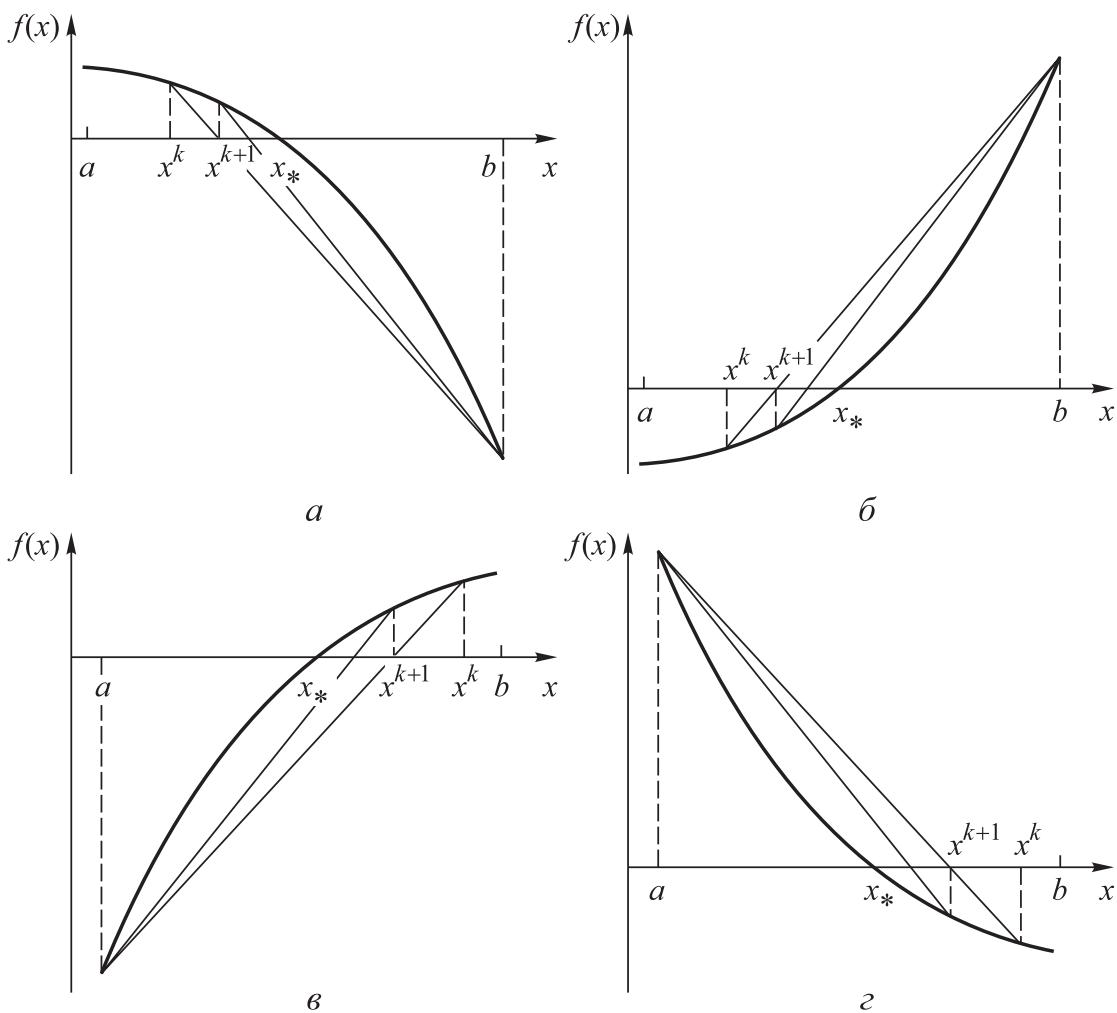


Рис. 5.4. Варианты метода секущих:
 $a, б$ — при $f'(x)f''(x) > 0$; $в, г$ — при $f'(x)f''(x) < 0$

функции $f(x)$, проходящая через точки $(x^k, f(x^k))$ и $(b, f(b))$, и за очередное приближение x^{k+1} к корню принимается абсцисса точки пересечения секущей с осью абсцисс (рис. 5.4).

Исследуем сходимость данного варианта метода хорд для случая дважды непрерывно дифференцируемой на отрезке $[a, b]$ функции $f(x)$ такой, что $f'(x) > 0$ и $f''(x) \geq 0$ (см. рис. 5.4, б). Пусть, как и ранее, x_* — искомый корень уравнения $f(x) = 0$ ($f(x_*) = 0$). Предположим, что на некоторой k -й итерации выполнено условие $x^k < x_* < b$.

Исследуем значение x^{k+1} . В соответствии с используемым итерационным процессом

$$\begin{aligned} x^{k+1} - x^k &= -f(x^k) \frac{b - x^k}{f(b) - f(x^k)} = \\ &= -[f(x^k) - f(x_*)] \frac{b - x^k}{f(b) - f(x_*) + f(x_*) - f(x^k)} = \\ &= -f'(\xi)(x^k - x_*) \frac{b - x^k}{f'(\eta)(b - x_*) + f'(\xi)(x_* - x^k)}. \end{aligned}$$

Преобразования выполнены с помощью формулы Лагранжа конечных приращений. Согласно принятым предположениям, фигурирующие в ней средние точки таковы: $\xi \in (x^k, x_*)$, $\eta \in (x_*, b)$, $\xi < \eta$. Тогда, исходя из знака второй производной, $f'(\eta) \geq f'(\xi)$ и $x^{k+1} - x^k \leq x_* - x^k$. Кроме того, поскольку x^k лежит на оси абсцисс левее корня, то справедливо неравенство $x^{k+1} - x^k > 0$.

В результате получаем, что последовательность итерационных приближений $\{x^k\}$ монотонно возрастает и ограничена сверху значением корня. Следовательно, по теореме Вейерштрасса она сходится. Легко видеть, что ее пределом является искомый корень уравнения.

Точно так же работает метод хорд в случае функции $f(x)$ такой, что $f'(x) < 0$ и $f''(x) \leq 0$. Если же первая и вторая производные функции имеют разные знаки (см. рис. 5.4, в и г), то в качестве зафиксированной точки выбирается левая граница отрезка локализации $x = a$. Таким образом, при совпадении знаков производных корень лежит правее точки пересечения хорды с осью Ox (см. рис. 5.4, а и б), а при их различии — левее (см. рис. 5.4, в и г).

Пример 5.3. Решим методом хорд уравнение

$$f(x) = 1 - x^2 = 0.$$

Отрезок локализации корня $[0,1; 2]$, $b = 2$.

Результаты, сгенерированные итерационным процессом метода хорд для разных начальных приближений, приведены в табл. 5.2.

Таблица 5.2

Сходимость метода хорд для уравнения $1 - x^2 = 0$

x^k	Вариант 1	Вариант 2
x^0	2,0	0,1
x^1	1,25	0,571428571428571
x^2	1,07692307692308	0,8(3)
x^3	1,025	0,941176470588235
x^4	1,00826446280992	0,98
x^5	1,00274725274725	0,993288590604027
x^6	1,00091491308326	0,997757847533632
x^7	1,00030487804878	0,999252056843680
x^8	—	0,999750623441397

Отметим, что метод хорд сходится монотонно. •

Метод Ньютона (метод касательных). Для численного решения уравнения $f(x) = 0$ заменим $f(x)$ на $\tilde{f}(x)$ с помощью интерполяции Эрмита по одной точке, считая известными в ней значения функции и ее производной (т. е. воспользуемся формулой Тейлора). В таком случае прямая

$$\tilde{f}(x) = f(x^k) + f'(x^k)(x - x^k)$$

проходит через точку $(x^k, f(x^k))$, в которой известны и функция, и ее производная, т. е. $f(x)$ заменяется касательной к ее графику в точке $(x^k, f(x^k))$.

Получаем **метод Ньютона**:

$$x^{k+1} = x^k - f(x^k)/f'(x^k).$$

Это один из наиболее эффективных, быстрых методов, однако для его применения требуются особые условия.

Исследуем условия его сходимости с помощью представления в виде метода типа простой итерации. В этом случае

$$F(x) = x - f(x)/f'(x).$$

Тогда

$$F'(x) = 1 - f f''/(f')^2 = f f''/(f')^2.$$

Пусть всюду на отрезке $[a, b]$

$$|f'(x)| \geq m > 0; \quad |f''(x)| \leq M.$$

Тогда существует такая ε -окрестность корня x_* , что если начальное приближение $x^0 \in \tilde{O}_\varepsilon(x_*)$, то итерационный процесс сходится к корню.

Действительно, всюду на отрезке $[a, b]$

$$|F'(x)| = \left| \frac{f f''}{(f')^2} \right| \leq \frac{|f|}{m^2} M.$$

Из непрерывности функции $f(x)$ следует, что для любого $q \in (0; 1)$ в некоторой $\varepsilon(q)$ -окрестности корня x_* справедливо неравенство

$$|f(x)| \leq q m^2 / M.$$

Таким образом, в этой окрестности выполнены условия следствия 5.1 и справедливы выводы теоремы 5.1.

Следовательно, речь идет о сходимости метода Ньютона в малом, т. е. лишь при удачном попадании начального приближения в окрестность корня. Это и неудивительно, ведь, по сути, метод Ньютона основан на экстраполяции функции $f(x)$.

Оценим погрешность метода Ньютона $x^{k+1} = x^k - f(x^k)/f'(x^k)$. Разложим $f(x_*)$ по формуле Тейлора с центром в точке x^k :

$$f(x_*) = f(x^k) + f'(x^k)(x_* - x^k) + \frac{1}{2} f''(\xi)(x^k - x_*)^2 = 0.$$

Здесь $\xi \in (x^k, x_*)$ или $\xi \in (x_*, x^k)$ в зависимости от взаимного расположения корня и его приближения.

Запишем выражение для x^{k+1} в виде

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

Следовательно,

$$|x^{k+1} - x_*| \leq \frac{M}{2m} |x^k - x_*|^2 \leq \left(\frac{M}{2m} \right)^{2^{k+1}-1} |x^0 - x_*|^{2^{k+1}}.$$

Можно получить более строгие оценки, если предположить, что функция $f(x)$ имеет монотонную производную $f'(x)$ определенного знака на отрезке $[a, b]$.

Метод Ньютона имеет квадратичную скорость сходимости, если $f'(x) \neq 0$, что видно из последнего неравенства. В противном случае скорость сходимости снижается до линейной.

Говорят, что **метод сходится с p -м порядком**, если погрешность метода удовлетворяет оценке

$$|x^{k+1} - x_*| \leq L |x^k - x_*|^p.$$

Замечание 5.4. Существуют многочисленные варианты метода Ньютона. Самый простой из них заключается в использовании во всех итерациях производной, вычисленной в какой-то одной точке. При этом квадратичная скорость сходимости теряется.

Замечание 5.5. Рассмотрим следующую модификацию метода Ньютона:

$$x^{k+1} = x^k - \frac{f(x^k)}{f'(x^k)} - \frac{f\left(x^k - f(x^k)f'(x^k)^{-1}\right)}{f'(x^k)}.$$

Нетрудно видеть, что скорость сходимости данного метода — кубическая. Этому способствует добавленный «подшаг» из нового ньютоновского приближения вдоль прямой, имеющей тот же угол наклона, что и касательная в x^k .

Пример 5.4. Найдем методом Ньютона корень уравнения $f(x) = 1 - x^2 = 0$ на отрезке $[0,1; 2]$.

Положим, что $x^0 = 2$, и организуем итерационный процесс:

$$\begin{aligned}x^1 &= x^0 - \frac{f(x^0)}{f'(x^0)} = 2 - \frac{1 - 2^2}{-2 \cdot 2} = 1,25; \\x^2 &= x^1 - \frac{f(x^1)}{f'(x^1)} = 1,25 - \frac{1 - 1,25^2}{-2 \cdot 1,25} = 1,025; \\x^3 &= x^2 - \frac{f(x^2)}{f'(x^2)} = 1,025 - \frac{1 - 1,025^2}{-2 \cdot 1,025} \approx 1,0003.\end{aligned}$$

Таким образом, уже на третьей итерации погрешность не превышает $\varepsilon = 10^{-3}$ (сравните с методом релаксации, рассмотренным в примере 5.2).

Возьмем теперь в качестве начального приближения $x^0 = 0,1$, тогда

$$x^1 = x^0 - \frac{f(x^0)}{f'(x^0)} = 0,1 - \frac{1 - 0,1^2}{-2 \cdot 0,1} = 5,05.$$

Однако $x^1 \notin [0,1; 2]$, т. е. произошел выход за границы отрезка локализации корня. Отметим, что при использовании метода хорд для решения того же уравнения $1 - x^2 = 0$ (см. пример 5.3) выхода за границы отрезка локализации не произошло в силу построения этого метода.

Метод Ньютона имеет второе название — *метод касательных*, так как его можно интерпретировать следующим образом. На k -й итерации строится касательная к графику функции $f(x)$ в точке x^k и за следующее приближение к корню принимается абсцисса точки пересечения касательной с осью абсцисс (рис. 5.5). В рассматриваемом примере эта точка оказалась за пределами отрезка локализации, поскольку касательные к $f(x)$ в точках $x \rightarrow 0$ практически параллельны оси абсцисс (так как $f'(x) \rightarrow 0$ при $x \rightarrow 0$). В этом случае следует модифицировать алгоритм решения либо искать другое начальное приближение.

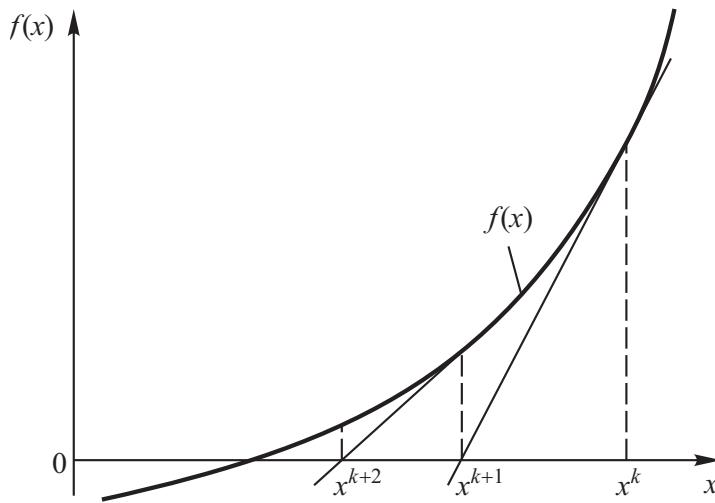


Рис. 5.5. Геометрическая интерпретация сходимости метода Ньютона

Иногда метод Ньютона сходится не к искомому корню, а к локальному минимуму, поэтому на практике имеет смысл проверить близость $f(x^{k_0})$ к нулю. •

Пример 5.5. Найдем решение уравнения

$$f(x) = 35x^3 - 67x^2 - 3x + 3 = 0.$$

На отрезке $[0; 1]$ существует единственный корень $x_* = 0,2$. Попробуем найти его методом Ньютона при $x^0 = 0$:

$$x^1 = x^0 - \frac{f(x^0)}{f'(x^0)} = 1;$$

$$x^2 = x^1 - \frac{f(x^1)}{f'(x^1)} = 0,$$

т. е. $x^2 = x^0$, произошло так называемое зацикливание. Если изменить начальное приближение, то итерационный процесс сойдется. Например, при $x^0 = 0,5$ уже на четвертой итерации $|x^4 - x_*| < 3 \cdot 10^{-6}$. •

Пример 5.6. В условиях сходимости метода Ньютона есть условие ограничения производной $f'(x)$ на отрезке локализации

корня $|f'(x)| > 0$. Если $f'(x_*) = 0$, то x_* — кратный корень. В этом случае метод Ньютона сходится лишь с линейной скоростью. Для того чтобы увидеть разницу между линейной и квадратичной скоростью сходимости, рассмотрим два уравнения:

$$f_1(x) = x^2 - 1 = 0;$$

$$f_2(x) = (x - 1)^2 = 0.$$

Применим для их решения метод Ньютона, начиная в обоих случаях с $x^0 = 2$. Результаты решения приведены в таблице 5.3.

Таблица 5.3

Сходимость метода Ньютона к простому и к кратному корню

x^k	Простой корень $f_1(x) = x^2 - 1 = 0$	Кратный корень $f_2(x) = (x - 1)^2 = 0$
x^0	2,0	2,0
x^1	1,25	1,5
x^2	1,025	1,25
x^3	1,000304878080488	1,125
x^4	1,0000000464611	1,0625
x^5	1,0	1,03125

Видно, что в случае уравнения $f_2(x) = 0$ метод Ньютона сходится гораздо медленнее, так как $f'_2(x_*) = 0$. •

Метод парабол. Построим интерполяционный полином второго порядка, воспользовавшись для этого тремя точками. Такой полином может быть построен в виде

$$\tilde{f}(x) = ax^2 + bx + c,$$

где a, b, c определяются из системы уравнений

$$\tilde{f}(x^i) = a(x^i)^2 + bx^i + c = f(x^i), \quad i = k-2, k-1, k.$$

С помощью формы Лагранжа интерполяционного полинома уравнение этой параболы можно записать без непосредственного решения последней системы в виде

$$\begin{aligned}\tilde{f}(x) = & f(x^{k-2}) \frac{(x - x^{k-1})(x - x^k)}{(x^{k-2} - x^{k-1})(x^{k-2} - x^k)} + \\ & + f(x^{k-1}) \frac{(x - x^{k-2})(x - x^k)}{(x^{k-1} - x^{k-2})(x^{k-1} - x^k)} + \\ & + f(x^k) \frac{(x - x^{k-2})(x - x^{k-1})}{(x^k - x^{k-2})(x^k - x^{k-1})}.\end{aligned}$$

Тогда приближение к корню x^{k+1} принимается равным одному из корней уравнения $\tilde{f}(x) = 0$. Как правило, выбирают тот корень, который ближе к x^k . Получаемый метод называется **методом парабол**.

Несмотря на повышение порядка приближающей функции, что, казалось бы, должно привести к повышению скорости сходимости, выигрыш здесь очень небольшой. Скорость сходимости данного метода ниже квадратичной, но выше скорости сходимости метода секущих.

Метод парабол принципиально отличается от методов, основанных на замене исходной функции ее линейным приближением, тем, что может дать комплексные корни при действительных предыдущих приближениях. Отметим также, что метод парабол оказывается весьма эффективным средством нахождения корней алгебраических многочленов. Итерационные приближения данного метода практически всегда быстро сходятся к корню уравнения.

Интерполяция Эрмита второго порядка. Если для численного решения уравнения $f(x) = 0$ воспользоваться аналогично методу Ньютона интерполяцией Эрмита, считая при этом, что $f''(x^k)$ легко вычислена, придем к другой параболической аппроксимации функции $f(x)$. Вместо функции $f(x)$ используем полином Тейлора второго порядка

$$\tilde{f}(x) = f(x^k) + f'(x^k)(x - x^k) + \frac{1}{2}f''(x^k)(x - x^k)^2,$$

проходящей через точку x^k , в которой известны сама функция, ее первая и вторая производные. Тогда

$$x^{k+1} = x^k - \frac{f'(x^k)}{f''(x^k)} \left[1 \pm \left(1 - 2 \frac{f(x^k)f''(x^k)}{(f'(x^k))^2} \right)^{1/2} \right].$$

Разложив подкоренное выражение с точностью до квадратичных слагаемых и выбрав соответствующий знак, получим *модификацию метода Ньютона*, дающую кубическую скорость сходимости:

$$x^{k+1} = x^k - \frac{f(x^k)}{f'(x^k)} - \frac{f''(x^k) [f(x^k)]^2}{2 [f'(x^k)]^3}.$$

Можно этого не делать и использовать предыдущее выражение для корней непосредственно. В этом случае будем иметь еще один вариант *метода типа парабол*.

Замечание 5.6. Инструментарий исследования сходимости на основе теоремы 5.1 применим только к одношаговым (т. е. связывающим только две последовательные итерации) методам. К ним не относится ни метод парабол, ни метод хорд в общей постановке.

5.2. Решение систем нелинейных уравнений

5.2.1. Постановка задачи и основные понятия

Рассмотрим систему уравнений

$$f_i(x_1, x_2, \dots, x_n) = 0, \quad i = \overline{1, n},$$

и поставим задачу поиска n неизвестных x_1, x_2, \dots, x_n . Будем полагать, что x есть n -мерный вектор, компоненты которого равны x_1, x_2, \dots, x_n , $x \in \mathbb{R}^n$, а $F(x) = (f_1(x), f_2(x), \dots, f_n(x))^\top$ — отображение пространства \mathbb{R}^n в \mathbb{R}^n .

Тогда систему уравнений можно записать в операторном виде:

$$F(x) = 0.$$

Рассмотрим итерационные одношаговые методы решения такого уравнения, имеющие вид

$$B_{k+1} \frac{x^{k+1} - x^k}{\tau_{k+1}} + F(x^k) = 0, \quad k = 0, 1, \dots,$$

где B_{k+1} — невырожденная матрица, определяющая метод; k — номер итерации; $x^k = (x_1^k, x_2^k, \dots, x_n^k)^T$ — значение приближенного решения на k -й итерации; τ_{k+1} — числовой параметр. Существуют и нелинейные (относительно x^{k+1}) итерационные методы. Ограничимся пока методами указанного вида.

Для вычисления x^{k+1} необходимо решить линейное операторное уравнение

$$B_{k+1}x^{k+1} = g(x^k) = B_{k+1}x^k - \tau_{k+1}F(x^k).$$

Как и ранее, метод называется *явным*, если $B_{k+1} = E$, и *неявным* в противном случае, *стационарным* при $B_{k+1} = B$, $\tau_{k+1} = \tau$ и *нестационарным* в противном случае.

Определение. При решении системы уравнений

$$B_{k+1}x^{k+1} = g(x^k)$$

итерационным способом *итерации* последнего называются *внутренними*, а *итерации* x^k — *внешними*.

5.2.2. Сходимость стационарного метода

Рассмотрим стационарный одношаговый итерационный метод

$$B \frac{x^{k+1} - x^k}{\tau} + F(x^k) = 0.$$

Тогда

$$x^{k+1} = x^k - \tau B^{-1}F(x^k) = S(x^k).$$

Исходное уравнение $F(x) = 0$ можно переписать в виде $x = S(x)$,

в котором $\tau \neq 0$, матрица B невырождена. Искомое решение при этом является неподвижной точкой оператора $S(x)$.

Определение. *Оператор S называется **сжимающим** на множестве K с коэффициентом сжатия q , если существует такое число $q \in (0; 1)$, что*

$$\|S(x') - S(x'')\| \leq q \|x' - x''\|, \quad x', x'' \in K.$$

Теорема 5.2 (принцип сжимающих отображений). Пусть оператор S определен в шаре радиусом r

$$U_r(a) = \{x : \|x - a\| \leq r\}$$

и является сжимающим в нем с коэффициентом q , причем

$$\|S(a) - a\| \leq (1 - q)r, \quad q \in (0; 1).$$

Тогда в $U_r(a)$ оператор S имеет единственную неподвижную точку x_* , итерации $x^{k+1} = S(x^k)$ сходятся к x_* для любого $x^0 \in U_r(a)$. Для погрешности справедливы оценки

$$\|x^k - x_*\| \leq q^k \|x^0 - x_*\|; \quad \|x^k - x_*\| \leq \frac{q^k}{1 - q} \|S(x^0) - x^0\|.$$

◀ Пусть $x^0 \in U_r(a)$ и все $x^k \in U_r(a)$. Докажем, что и $x^{k+1} \in U_r(a)$.

Действительно,

$$x^{k+1} - a = S(x^k) - a = S(x^k) - S(a) + S(a) - a.$$

Тогда

$$\begin{aligned} \|x^{k+1} - a\| &\leq \|S(x^k) - S(a)\| + \|S(a) - a\| \leq \\ &\leq q \|x^k - a\| + (1 - q)r \leq qr + (1 - q)r = r \end{aligned}$$

и, следовательно, $x^{k+1} \in U_r(a)$.

Оценим $x^{k+1} - x^k$:

$$\begin{aligned} \|x^{k+1} - x^k\| &= \|S(x^k) - S(x^{k-1})\| \leq \\ &\leq q \|x^k - x^{k-1}\| \leq q^k \|S(x^0) - x^0\|. \end{aligned}$$

Отсюда можно заключить, что последовательность $\{x^k\}$ является фундаментальной, а именно:

$$\begin{aligned} \|x^{k+p} - x^k\| &= \left\| \sum_{l=1}^p (x^{k+l} - x^{k+l-1}) \right\| \leqslant \\ &\leqslant \|S(x^0) - x^0\| \sum_{l=1}^p q^{k+l-1} = q^k \frac{1 - q^p}{1 - q} \|S(x^0) - x^0\| \leqslant \\ &\leqslant \frac{q^k}{1 - q} \|S(x^0) - x^0\|. \end{aligned}$$

Поскольку множество $U_r(a)$ замкнуто, то $\{x^k\} \rightarrow x_* \in U_r(a)$. Оператор S — непрерывный в $U_r(a)$ в силу сжимаемости, поэтому, переходя к пределу при $k \rightarrow \infty$ в выражении $x^{k+1} = S(x^k)$, получим $x_* = S(x_*)$, т. е. x_* — решение уравнения $F(x) = 0$. Следовательно, решение в $U_r(a)$ существует. Оно единственно, так как, допустив существование второго решения x_{**} , имеем

$$\|x_* - x_{**}\| = \|S(x_*) - S(x_{**})\| \leqslant q \|x_* - x_{**}\|.$$

Учитывая, что $q \in (0; 1)$, приходим к следующему заключению: $\|x_* - x_{**}\| = 0$, т. е. $x_* = x_{**}$.

Переходя к пределу при $p \rightarrow \infty$ в неравенстве для $\|x^{k+p} - x^k\|$, получаем

$$\|x_* - x^k\| \leqslant \frac{q^k}{1 - q} \|S(x^0) - x^0\|,$$

$$\|x^{k+1} - x_*\| = \|S(x^k) - S(x_*)\| \leqslant q \|x^k - x_*\| \leqslant q^{k+1} \|x^0 - x_*\|,$$

т. е. оценки погрешности решения. ►

Замечание 5.7. По существу, теорема 5.1 есть частный случай теоремы 5.2.

5.2.3. Примеры итерационных методов

При решении системы уравнений $F(x) = 0$ используют различные итерационные методы, такие как методы релаксации, Пикара, Ньютона, Якоби, Зейделя, а также гибридные методы. Рассмотрим итерационные методы вида

$$B_{k+1} \frac{x^{k+1} - x^k}{\tau_{k+1}} + F(x^k) = 0, \quad k = 0, 1, \dots.$$

Метод релаксации. Этому методу соответствует выбор $B_{k+1} = E$, $\tau_{k+1} = \tau$, $S(x) = x - \tau F(x)$. Метод сходится, если норма $\|S'\| < 1$, где $S' = E - \tau F'$ и матрица Якоби

$$F' = \begin{pmatrix} (f_1)'_{x_1} & (f_1)'_{x_2} & \cdots & (f_1)'_{x_n} \\ \cdots & \cdots & \cdots & \cdots \\ (f_n)'_{x_1} & (f_n)'_{x_2} & \cdots & (f_n)'_{x_n} \end{pmatrix}.$$

Метод Пикара. Пусть $F(x) = Ax + G(x)$, где A — линейный оператор; $G(x)$ — некоторая вектор-функция. Тогда итерации можно определить следующим образом:

$$Ax^{k+1} + G(x^k) = 0,$$

т. е. $B_{k+1} = A$; $\tau_{k+1} = \tau = 1$; $S(x) = x - \tau B^{-1}F = x - A^{-1}F$.

Метод сходится при $\|S'\| = \|E - A^{-1}F'\| < 1$.

Можно провести **модификацию метода Пикара**: вместо $A(x^{k+1} - x^k) + F(x^k) = 0$ записать

$$A \frac{x^{k+1} - x^k}{\tau} + F(x^k) = 0,$$

т. е. ввести параметр τ , который управляет скоростью сходимости.

Метод Ньютона. Возьмем $B_{k+1} = F'(x^k)$, $\tau_{k+1} = 1$, т. е.

$$F'(x^k)(x^{k+1} - x^k) + F(x^k) = 0.$$

Для реализации метода Ньютона необходимо существование матрицы, обратной матрице $F'(x^k)$.

Как и в скалярном случае, метод Ньютона имеет квадратичную сходимость, если начальное приближение выбрано удачно. Доказательство сходимости опустим.

Отметим, что приведенное уравнение относительно x^{k+1} , как и в случае скалярного уравнения, можно получить построением интерполянта Эрмита или усечением разложения $F(x)$ по формуле Тейлора в многомерном случае. Для его решения можно применить любой уже рассмотренный метод решения СЛАУ.

Модифицированный метод Ньютона. Упростим вычисления по методу Ньютона, зафиксировав значение матрицы Якоби в нулевом приближении: $B_{k+1} = F'(x^0)$; $\tau_{k+1} = 1$. При этом получать матрицу, обратную матрице B_{k+1} (или, например, вычислять LU -разложение), в отличие от исходного варианта метода нужно лишь один раз.

Метод Ньютона с параметром. Этот вариант метода имеет вид

$$F'(x^k) \frac{x^{k+1} - x^k}{\tau_{k+1}} + F(x^k) = 0.$$

Дополнительное исследование проводить не будем.

Приведем несколько примеров использования метода Ньютона, поскольку это один из наиболее распространенных методов решения нелинейных систем уравнений и задач минимизации.

Пример 5.7. Решим методом Ньютона систему уравнений

$$x_1 + x_2 = 3;$$

$$x_1^2 + x_2^2 = 9.$$

Корнями здесь являются $(3; 0)$ и $(0; 3)$. Для данной системы

$$F(x) = \begin{pmatrix} x_1 + x_2 - 3 \\ x_1^2 + x_2^2 - 9 \end{pmatrix}; \quad F'(x) = \begin{pmatrix} 1 & 1 \\ 2x_1 & 2x_2 \end{pmatrix}.$$

Пусть $x^0 = (1; 5)$, тогда две первые итерации метода Ньютона записываются следующим образом:

первая итерация

$$F'(x^0)(x^1 - x^0) = -F(x^0):$$

$$\begin{pmatrix} 1 & 1 \\ 2 & 10 \end{pmatrix} \begin{pmatrix} x_1^1 - x_1^0 \\ x_2^1 - x_2^0 \end{pmatrix} = - \begin{pmatrix} 3 \\ 17 \end{pmatrix} \Rightarrow \begin{pmatrix} x_1^1 - x_1^0 \\ x_2^1 - x_2^0 \end{pmatrix} = \begin{pmatrix} -1,625 \\ -1,375 \end{pmatrix};$$

$$x_1^1 = -0,625, \quad x_2^1 = 3,625;$$

вторая итерация

$$F'(x^1)(x^2 - x^1) = -F(x^1):$$

$$\begin{pmatrix} 1 & 1 \\ -1,25 & 7,25 \end{pmatrix} \begin{pmatrix} x_1^2 - x_1^1 \\ x_2^2 - x_2^1 \end{pmatrix} = - \begin{pmatrix} 0 \\ 4,53125 \end{pmatrix} \Rightarrow$$

$$\Rightarrow \begin{pmatrix} x_1^2 - x_1^1 \\ x_2^2 - x_2^1 \end{pmatrix} = \begin{pmatrix} 0,533088 \\ -0,533088 \end{pmatrix};$$

$$x_1^2 = -0,092; \quad x_2^2 = 3,092.$$

В данном случае метод Ньютона сходится быстро, уже на второй итерации получаем достаточно точное приближение к $(0; 3)^T$. Здесь проявилось основное преимущество метода Ньютона: если x^0 близко к решению x_* и якобиан $F'(x_*)$ невырожден, то скорость сходимости к x_* квадратичная. •

Пример 5.8. Применим метод Ньютона для решения системы уравнений

$$e^{x_1} = 1;$$

$$e^{x_2} = 1,$$

решение которой $(0; 0)^T$. Пусть $x^0 = (-10; -10)^T$, тогда $x^1 = (-11 + e^{10}; -11 + e^{10})^T$. Такое приближение сложно назвать приемлемым. Особенности сходимости метода Ньютона указывают на то, что при решении многомерных задач его следует применять

по крайней мере на заключительных итерациях, с тем чтобы использовать его быструю локальную сходимость.

Таким образом, недостатки метода Ньютона (отсутствие глобальной сходимости для многих задач, трудоемкость вычисления якобиана на каждой итерации, необходимость решать на каждой итерации СЛАУ, которая может быть вырожденной или плохо обусловленной) компенсируются квадратичной скоростью сходимости при достаточно точном приближении к корню. •

Рассмотренные методы линейны относительно x^{k+1} . Далее приведем несколько алгоритмов, в которых для вычисления x^{k+1} придется решать нелинейные системы уравнений.

Нелинейный метод Якоби. Для решения системы $F(x) = 0$ нелинейный метод Якоби определяется следующими формулами:

$$f_i(x_1^k, x_2^k, \dots, x_{i-1}^k, x_i^{k+1}, x_{i+1}^k, \dots, x_n^k) = 0, \quad i = \overline{1, n}.$$

При этом необходимо решить n скалярных, возможно нелинейных, уравнений, независимых друг от друга. Порядок решения уравнений может быть произвольным.

Нелинейный метод Зейделя. В этом методе новое итерационное приближение находится из уравнений

$$f_i(x_1^{k+1}, x_2^{k+1}, \dots, x_{i-1}^{k+1}, x_i^{k+1}, x_{i+1}^k, \dots, x_n^k) = 0, \quad i = \overline{1, n}.$$

При реализации нелинейного метода Зейделя необходимо решить n скалярных уравнений, но при решении i -го уравнения используется информация, полученная при решении предыдущих $i - 1$ уравнений.

Гибридные методы. Нелинейные методы Якоби и Зейделя на каждой итерации приводят к задаче последовательного решения скалярных нелинейных уравнений относительно x_i^{k+1} . Для решения каждого из этих уравнений снова придется использовать какой-либо итерационный метод. Это могут быть методы Ньютона, хорд, релаксации, «вилки» и др. Такая

комбинация методов дает **гибридный метод** решения системы нелинейных уравнений.

Рассмотрим примеры конструирования гибридных методов.

Пример 5.9. Построим гибридный метод решения системы нелинейных уравнений, используя внешние итерации по *методу Зейделя*, а внутренние — по *методу Ньютона*.

Применим метод Зейделя для решения системы нелинейных уравнений

$$f_i(x_1, x_2, \dots, x_n) = 0, \quad i = \overline{1, n}.$$

Из первого уравнения системы найдем x_1^{k+1} , беря значения остальных неизвестных с предыдущей итерации x_j^k , $j = \overline{2, n}$. Решив соответствующее скалярное нелинейное уравнение, подставим результат во все остальные уравнения. Второе уравнение системы перепишем как скалярное относительно переменной x_2^{k+1} . Решив его, подставим найденное значение в остальные уравнения и перейдем к решению третьего уравнения, и так до последнего уравнения системы. На каждой итерации по методу Зейделя систему уравнений можно записать в следующем виде:

$$f_i(x_1^{k+1}, x_2^{k+1}, \dots, x_{i-1}^{k+1}, x_i^{k+1}, x_{i+1}^k, \dots, x_n^k) = g_i(x_i^{k+1}) = 0, \quad i = \overline{1, n},$$

где k — номер внешней итерации по методу Зейделя.

Для решения скалярных уравнений относительно $\xi_i = x_i^{k+1}$ применим метод Ньютона:

$$g'_i(\xi_i^m)(\xi_i^{m+1} - \xi_i^m) + g_i(\xi_i^m) = 0, \quad i = \overline{1, n},$$

где m — номер внутренних итераций по методу Ньютона.

Внутренние итерации не всегда выполняются до сходимости, на практике иногда ограничиваются некоторым количеством M внутренних итераций, ξ_i^M принимают за x_i^{k+1} и проверяют сходимость только внешних итераций. В качестве начального приближения на внутренних итерациях принимают $\xi_i^0 = x_i^k$.

Алгоритм решения системы нелинейных уравнений, если внешние итерации организованы по методу Зейделя, а внутренние — по методу Ньютона, можно описать следующим образом.

Выбираем начальное приближение

для внешних итераций (x_1^0, \dots, x_n^0) .

Выполняем цикл по $k = 0, 1, 2, \dots$.

Вводим замену $\xi_i = x_i^{k+1}$.

Выполняем цикл по i от 1 до n .

Определяем функции

$$g_i(\xi_i) = f(x_1^{k+1}, \dots, x_{i-1}^{k+1}, \xi_i, x_{i+1}^k, \dots, x_n^k).$$

Выбираем начальное приближение

для внутренних итераций $\xi_i^0 = x_i^k$.

Выполняем цикл по $m = 0, 1, 2, \dots$.

Решаем линейные уравнения

$$g_i'(\xi_i^m)(\xi_i^{m+1} - \xi_i^m) + g_i(\xi_i^m) = 0.$$

Если $|\xi_i^{m+1} - \xi_i^m| < \varepsilon_1$ или $m + 1 > M$,

то цикл останавливается.

Присваиваем $x_i^{k+1} = \xi_i^{m+1}$.

Если $\|x^{k+1} - x^k\| < \varepsilon_2$ или $k > N$,

то цикл останавливается.

В итоге гибридный метод, в котором внешние итерации выполняются по методу Зейделя, а внутренние — по методу Ньютона, можно записать в следующем виде:

$$\begin{aligned} \frac{\partial f_i}{\partial x_i}(x_1^{k+1}, \dots, x_{i-1}^{k+1}, (x_i^{k+1})^m, x_{i+1}^k, \dots, x_n^k) \times \\ \times \left[(x_i^{k+1})^{m+1} - (x_i^{k+1})^m \right] + \\ + f_i(x_1^{k+1}, \dots, x_{i-1}^{k+1}, (x_i^{k+1})^m, x_{i+1}^k, \dots, x_n^k) = 0, \end{aligned}$$

где k — номер внешней итерации; m — номер внутренней итерации.

Если выполнить всего одну внутреннюю итерацию ($M = 1$), приняв ее результат за x_i^{k+1} , то получим некий новый метод. В случае $n = 2$ он имеет вид

$$\frac{\partial f_1}{\partial x_1}(x_1^k, x_2^k)(x_1^{k+1} - x_1^k) + f_1(x_1^k, x_2^k) = 0;$$

$$\frac{\partial f_2}{\partial x_2}(x_1^{k+1}, x_2^k)(x_2^{k+1} - x_2^k) + f_2(x_1^{k+1}, x_2^k) = 0.$$

Выражения для x_1^{k+1} и x_2^{k+1} можно легко записать в явной форме. •

Пример 5.10. Построим гибридный метод решения системы нелинейных уравнений, используя внешние итерации по методу Ньютона, а внутренние — по методу Зейделя.

Внешние итерации метода Ньютона имеют вид

$$F'(x^k)(x^{k+1} - x^k) + F(x^k) = 0,$$

где

$$F' = A_- + D + A_+$$

(A_- , A_+ — нижняя и верхняя треугольные матрицы соответственно; D — диагональная матрица).

В данном методе для решения СЛАУ, полученной методом Ньютона на внешней итерации, применяется внутренний итерационный процесс — метод Зейделя:

$$(A_- + D)(x^{k+1})^{m+1} + A_+(x^{k+1})^m - F'(x^k)x^k + F(x^k) = 0,$$

где k — номер внешней итерации; m — номер внутренней итерации.

Если снова выполнить только одну внутреннюю итерацию, то получим новый (по существу, явный) метод.

Для случая $n = 2$ этот метод имеет вид

$$\begin{aligned} \frac{\partial f_1}{\partial x_1}(x_1^k, x_2^k)(x_1^{k+1} - x_1^k) + f_1(x_1^k, x_2^k) = 0; \\ \frac{\partial f_2}{\partial x_1}(x_1^k, x_2^k)(x_1^{k+1} - x_1^k) + \frac{\partial f_2}{\partial x_2}(x_1^k, x_2^k)(x_2^{k+1} - x_2^k) + \\ + f_2(x_1^k, x_2^k) = 0. \end{aligned}$$

Данный метод отличается от рассмотренного в предыдущем примере. •

Два последних примера демонстрируют сравнительную легкость конструирования новых итерационных методов для решения систем нелинейных уравнений. Отметим, что применение метода Ньютона на внешних итерациях приводит к решению СЛАУ, которую можно решать различными методами, как прямыми (методы Гаусса, Холецкого, QR-метод и др.), так и итерационными (методы Якоби, Зейделя, минимальных невязок, скорейшего спуска и др.).

Вопросы и задания

1. Приведите общий вид методов решения скалярных нелинейных уравнений типа простой итерации. Какой итерационный процесс называют сходящимся? Сформулируйте условия сходимости итерационных методов решения скалярных нелинейных уравнений типа простой итерации.
2. Сформулируйте алгоритм метода деления отрезка пополам. Какой критерий может быть использован для прекращения итераций? Чему равна скорость сходимости алгоритма, с какой точностью определен корень уравнения?
3. Приведите примеры методов интерполяционного типа для решения скалярных нелинейных уравнений.

4. Сформулируйте алгоритм решения скалярного нелинейного уравнения методом Ньютона. Какой критерий прекращения итераций может быть использован? Как выбрать начальное приближение? Оцените скорость сходимости метода Ньютона. Можно ли назвать метод Ньютона методом простой итерации? Можно ли назвать метод Ньютона методом интерполяционного типа?
5. Сформулируйте алгоритм решения скалярного нелинейного уравнения методом хорд. Приведите схему доказательства сходимости метода.
6. Как можно увеличить скорость сходимости метода Ньютона?
7. В чем сходство и различия методов решения скалярных нелинейных уравнений и систем нелинейных уравнений?
8. Какие итерации называются внутренними, а какие внешними при решении систем нелинейных уравнений?
9. Приведите примеры линейных и нелинейных итерационных методов решения систем нелинейных уравнений.
10. Сформулируйте условия сходимости стационарного итерационного процесса решения систем нелинейных уравнений.
11. Приведите пример неравнозначности порядка выполнения внутренних и внешних итераций.

Библиографические комментарии

Современные задачи математической физики, как правило, нелинейны. Поэтому решать нелинейные уравнения и системы таких уравнений приходится очень часто. В связи с этим вопросы, связанные с решением таких уравнений, обсуждаются практически во всей учебной литературе, приведенной в библиографии.

Отметим особо книгу [61], посвященную решению больших систем нелинейных уравнений. Она содержит как теоретический материал, так и ценные практические указания. Выделим также работу [29], посвященную двум важным и тесно связанным задачам: безусловной минимизации и решению нелинейных уравнений. Авторы рассматривают численные методы ньютоновского (квазиньютоновского) типа, уделяя большое внимание программной реализации численных методов, а также их конструированию.

Во времена маломощных ЭВМ особое внимание уделялось построению методов повышенной скорости сходимости для решения нелинейных уравнений. Например, в классическом руководстве [10] и [11] подробно рассмотрен алгоритм решения проблемы собственных значений путем нахождения нулей определителя матрицы $A - \lambda E$.

Книги [4, 15, 35] содержат большой материал, посвященный решению нелинейных уравнений.

6. МЕТОДЫ ЧИСЛЕННОГО ИНТЕГРИРОВАНИЯ И ДИФФЕРЕНЦИРОВАНИЯ

Представлены квадратурные формулы для численного нахождения одномерных и многомерных интегралов. Рассмотрены квадратурные формулы интерполяционного типа (включая формулы прямоугольников, трапеций, Симпсона и др.), квадратурные формулы Гаусса. Описаны способы вычисления несобственных интегралов I и II рода, интегралов от быстроосциллирующих функций. Приведены способы численного дифференцирования функций.

6.1. Простейшие квадратурные формулы

6.1.1. Постановка задачи и основные определения

Известно, что первообразные даже элементарных функций чаще всего не являются элементарными функциями. Поэтому аналитически вычислить определенный интеграл удается далеко не всегда. Приходится прибегать к численным методам.

Согласно определению интеграла Римана,

$$I = \int_a^b f(x) dx = \lim_{\lambda_R \rightarrow 0} \sum_{i=1}^N f(\xi_i) h_i,$$

где $\lambda_R = \max_{1 \leq i \leq N} h_i$; $h_i = x_i - x_{i-1} > 0$.

При этом набор точек $a = x_0 < x_1 < x_2 < \dots < x_N = b$, лежащих на отрезке $[a, b]$, называется разбиением R этого отрезка, λ_R — диаметром разбиения, точки $\xi_i \in [x_{i-1}, x_i]$ выбираются произвольно, как и точки разбиения.

Отсюда возникает идея приближенного вычисления интеграла I путем его замены суммой

$$I_h = \sum_{k=1}^n c_k f(\tilde{x}_k),$$

где n — конечное число.

Определение. Приближенное равенство $I \approx I_h = \sum_{k=1}^n c_k f(\tilde{x}_k)$ называется **квадратурной формулой**, I_h — **квадратурной суммой**, точки \tilde{x}_k — **узлами квадратурной суммы**, c_k — ее **весовыми коэффициентами (весами)**. **Погрешностью квадратурной формулы** называется разность $\psi_h = I - I_h$.

Довольно часто выражение для I_h также называется квадратурной формулой.

Будем считать, что функция f известна в узлах сетки

$$\tilde{\Omega}_h = \{\tilde{x}_i : a \leq \tilde{x}_1 < \tilde{x}_2 < \dots < \tilde{x}_n \leq b\}.$$

Иногда узлы сетки задают специальным образом, иногда — произвольным. Ограничимся пока равномерной сеткой с шагом

$$h_i = h = (b - a)/n.$$

При этом сетка

$$\Omega_h = \{x_i : x_i = a + ih, i = \overline{0, n}\}.$$

Для построения квадратурной формулы зачастую достаточно рассмотреть отрезок $[x_{i-1}, x_i]$, так как

$$I = \int_a^b f(x) dx = \sum_{i=1}^n \int_{x_{i-1}}^{x_i} f(x) dx = \sum_{i=1}^n I_i.$$

Тогда $I_h = \sum_{i=1}^n I_{h,i}$.

Простейшие квадратурные формулы возникают из рассмотренных ранее идей аппроксимации и интерполяции функции

на сетке (см. главу 4). Квадратурные формулы, для построения которых используются интерполяции, называются **квадратурными формулами интерполяционного типа**.

6.1.2. Формула прямоугольников

Для приближенного вычисления интеграла $\int_a^b f(x)dx$ воспользуемся кусочно-постоянной реконструкцией функции $f(x)$ на сетке. Тогда, если на отрезке $[x_{i-1}, x_i]$ в качестве значения функции взять $f(x_{i-1/2})$, где $x_{i-1/2} = x_{i-1} + h/2 = (x_i + x_{i-1})/2$, получим **квадратурную формулу центральных прямоугольников**

$$I_{h,i} = f(x_{i-1/2})h.$$

Напомним, что с геометрической точки зрения значение определенного интеграла есть площадь фигуры под кривой $y = f(x)$. В этом случае истинная криволинейная трапеция заменяется на прямоугольник, площадь которого равна $f(x_{i-1/2})h$ (рис. 6.1).

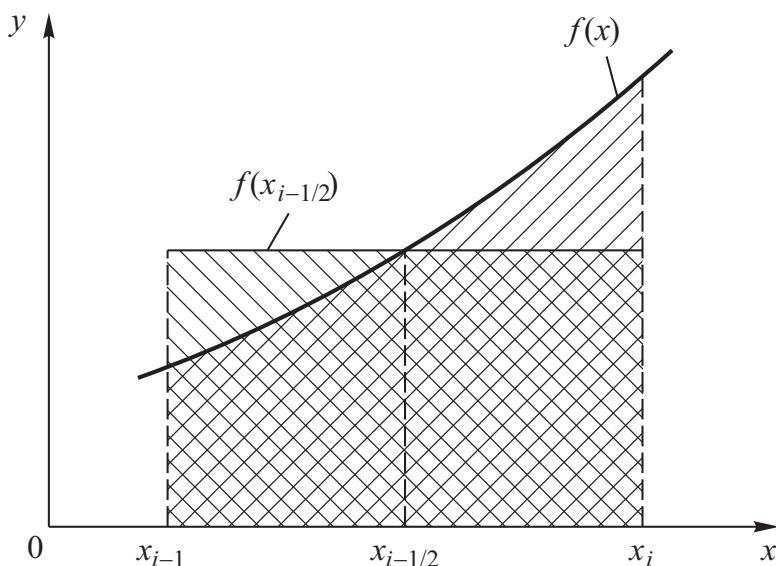


Рис. 6.1. Геометрическая интерпретация вычисления интеграла по формуле центральных прямоугольников

Погрешность квадратурной формулы

$$\psi_{h,i} = I_i - I_{h,i} = \int_{x_{i-1}}^{x_i} f(x) dx - I_{h,i}.$$

Если функция $f(x)$ дважды непрерывно дифференцируема на отрезке $[a, b]$, так что $\|f''\|_C \leq M_2$, то

$$\begin{aligned} |\psi_{h,i}| &= \left| \int_{x_{i-1}}^{x_i} [f(x) - f(x_{i-1/2})] dx \right| = \\ &= \left| \int_{x_{i-1}}^{x_i} [f(x_{i-1/2}) + f'(x_{i-1/2})(x - x_{i-1/2}) + \right. \\ &\quad \left. + \frac{1}{2}f''(x_{i-1/2}^*)(x - x_{i-1/2})^2 - f(x_{i-1/2})] dx \right| \leq \\ &\leq \frac{M_2}{2} \int_{x_{i-1}}^{x_i} (x - x_{i-1/2})^2 dx = M_2 \frac{h^3}{24}. \end{aligned}$$

Полученная оценка погрешности $|\psi_{h,i}|$ неулучшаемая: функция $f = (x - x_{i-1/2})^2$ реализует в данной оценке равенство $M_2 = 2$.

Для всего отрезка оценка погрешности квадратурной формулы $I_h = \sum_{i=1}^n f(x_{i-1/2})h$ имеет вид

$$|\psi_h| \leq M_2 \frac{h^3}{24} n = \frac{M_2}{24} \frac{(b-a)^3}{n^2} = O(h^2).$$

Если $\psi_h = O(h^l)$, то говорят, что **квадратурная формула** имеет **l -й порядок точности**. Для формулы центральных прямоугольников $l = 2$.

Если взять **формулы левых** ($I_{h,i} = f(x_{i-1})h$) или **правых** ($I_{h,i} = f(x_i)h$) **прямоугольников**, то в результате вычисления функции $f(x)$ в нецентральной точке в обоих случаях получим погрешность квадратурной формулы $\psi_h = O(h)$.

6.1.3. Формула трапеций

Для построения формулы трапеций заменим в интеграле $\int_a^b f(x)dx$ функцию $f(x)$ на отрезке $[x_{i-1}, x_i]$ линейным интерполянтом

$$\tilde{f}(x) = \frac{1}{h} [(x - x_{i-1})f(x_i) + (x_i - x)f(x_{i-1})].$$

Известно, что для дважды непрерывно дифференцируемых функций

$$|f - \tilde{f}| \leq \frac{1}{2} M_2 (x - x_{i-1})(x_i - x)$$

(см. замечание 4.5). Тогда интеграл от \tilde{f} дает **квадратурную формулу трапеций**

$$I_{h,i} = \frac{1}{2} [f(x_{i-1}) + f(x_i)] h$$

с оценкой локальной погрешности

$$\begin{aligned} |\psi_{h,i}| &\leq \frac{1}{2} M_2 \int_{x_{i-1/2}}^{x_i} (x - x_{i-1})(x_i - x) dx = \\ &= \frac{1}{2} M_2 \left[\frac{1}{2} (x - x_{i-1})^2 (x_i - x) + \frac{1}{6} (x - x_{i-1})^3 \right] \Big|_{x_{i-1}}^{x_i} = \\ &= \frac{1}{12} M_2 h^3. \end{aligned}$$

Отсюда получаем оценку погрешности квадратурной формулы трапеций

$$|\psi_h| \leq \frac{1}{2} M_2 h^2 (b - a) = O(h^2).$$

Оценки погрешности формул трапеций и центральных прямоугольников имеют один и тот же порядок, однако формула трапеций включает на одно вычисление функции больше.

6.1.4. Формула Симпсона

Еще одну квадратурную формулу для вычисления $\int_a^b f(x) dx$ можно получить уже известным способом — на сетке из трех узлов заменить подынтегральную функцию $f(x)$ квадратичным полиномом $L_2(x)$ и проинтегрировать его. Указанный интерполянт на равномерной сетке, состоящей из узлов x_{2i} , x_{2i+1} , x_{2i+2} , имеет вид

$$\begin{aligned} L_2(x) = & f_{2i} \frac{(x - x_{2i+1})(x - x_{2i+2})}{2h^2} - \\ & - f_{2i+1} \frac{(x - x_{2i})(x - x_{2i+2})}{h^2} + \\ & + f_{2i+2} \frac{(x - x_{2i})(x - x_{2i+1})}{2h^2}. \end{aligned}$$

Проинтегрировав интерполянт, получим так называемую **квадратурную формулу Симпсона**:

$$I_{h,i} = \int_{x_{2i}}^{x_{2i+2}} L_2(x) dx = \frac{h}{3} (f_{2i} + 4f_{2i+1} + f_{2i+2}).$$

Тогда, если вся сетка состоит из четного числа узлов $n = 2m$, интеграл по отрезку $[a, b]$ можно вычислить с помощью формулы

$$\int_a^b f(x) dx \approx I_h = \sum_{i=1}^m I_{h,i}.$$

Выведем ту же формулу иначе. Рассмотрим отрезок $[x_{2i}, x_{2i+2}]$ длиной $2h$. На нем имеются три точки сетки: x_{2i} , x_{2i+1} , x_{2i+2} . Приближенно вычислим интеграл $I_i = \int_{x_{2i}}^{x_{2i+2}} f(x) dx$. Пусть по *квадратурной формуле трапеций* величина

$$I_h^{(1)} = \frac{1}{2} [f(x_{2i}) + f(x_{2i+2})] \cdot 2h = h [f(x_{2i}) + f(x_{2i+2})]$$

построена по двум точкам x_{2i} , x_{2i+2} . При этом из оценки

погрешности квадратурной формулы трапеции (для $f \in C^{(4)}[a, b]$) имеем

$$I_h^{(1)} = \int_{x_{2i}}^{x_{2i+2}} f(x) dx + O(h^5) = I_i + C(2h)^3 + O(h^5).$$

Теперь учтем наличие трех точек. Дважды применив формулу трапеций на подынтервалах отрезка $[x_{2i}, x_{2i+2}]$, получим

$$\begin{aligned} I_h^{(2)} &= \frac{1}{2}[f(x_{2i}) + f(x_{2i+1})]h + \frac{1}{2}[f(x_{2i+1}) + f(x_{2i+2})]h = \\ &= I_i + 2Ch^3 + O(h^5). \end{aligned}$$

Из оценок погрешности интегрирования по отрезкам $[x_{2i}, x_{2i+1}]$, $[x_{2i+1}, x_{2i+2}]$ имеем

$$\begin{aligned} I_i &= \frac{1}{3}(4I_h^{(2)} - I_h^{(1)}) + O(h^5) = \\ &= \frac{h}{3}[f(x_{2i}) + 4f(x_{2i+1}) + f(x_{2i+2})] + O(h^5). \end{aligned}$$

Описанная процедура позволяет локально (на отрезке длиной $2h$) повысить порядок точности до пятого.

В приведенных выкладках использовалась формула Тейлора для значений функции $f(x)$ во всех точках с центром в точке x_{2i+1} :

$$\begin{aligned} f(x) &= f_{2i+1} + \frac{1}{1!}f'_{2i+1}(x - x_{2i+1}) + \frac{1}{2!}f''_{2i+1}(x - x_{2i+1})^2 + \\ &\quad + \frac{1}{3!}f'''_{2i+1}(x - x_{2i+1})^3 + \frac{1}{4!}\tilde{f}_{2i+1}^{(4)}(x - x_{2i+1})^4. \end{aligned}$$

Здесь $f_{2i+1}^{(k)} = \left. \frac{d^k f}{dx^k} \right|_{x=x_{2i+1}}$. В последнем слагаемом производная вычисляется в некоторой вспомогательной точке.

После интегрирования по отрезку $[x_{2i}, x_{2i+2}]$ слагаемые с нечетными степенями (нечетные функции относительно середины отрезка интегрирования) внесут нулевой вклад в интеграл.

Поэтому в результате останутся слагаемые $O(h^3)$ и $O(h^5)$. При этом требуется, чтобы функция имела четвертую непрерывную производную.

Таким образом, снова получена *квадратурная формула Симпсона*:

$$I_h = \frac{h}{3} [f(x_{2i}) + 4f(x_{2i+1}) + f(x_{2i+2})].$$

Процесс ее вывода с применением формулы трапеций на двух сетках ($2h$ и h — их шаги) называется *правилом Рунге*, иногда — *методом экстраполяции Ричардсона*.

Окончательно в случае использования формулы Симпсона по всему отрезку $[a, b]$ получим $\psi_h = O(h^4)$ для $f \in C^{(4)}[a, b]$.

6.2. Квадратурные формулы интерполяционного типа

Рассмотрим квадратурные формулы для вычисления интегралов вида

$$I = \int_a^b \rho(x)f(x) dx,$$

где $\rho(x) > 0$ — весовая функция; $\rho(x)$, $f(x)$ считаются достаточно гладкими функциями.

Как и ранее, квадратурные формулы имеют вид

$$I \approx \sum_{k=0}^n c_k f(x_k), \quad x_k \in [a, b],$$

где c_k — весовые коэффициенты.

Поставим задачу построения квадратурной формулы для вычисления интеграла I целиком по отрезку $[a, b]$ без разбиения на сумму интегралов по подотрезкам, как это делалось ранее. Рассмотрим сетку из $n + 1$ точек на отрезке $[a, b]$, вычислим

значения функции $f(x)$ в узлах сетки и заменим подынтегральную функцию на отрезке $[a, b]$ *интерполяционным многочленом Лагранжа* вида

$$L_n(x) = \sum_{k=0}^n f(x_k) \frac{\omega(x)}{(x - x_k)\omega'(x_k)},$$

$$\text{где } \omega(x) = \prod_{i=0}^n (x - x_i); \quad \omega'(x_k) = \prod_{\substack{i=0 \\ i \neq k}} (x_k - x_i).$$

Определение. Квадратурные формулы, полученные с помощью процедуры интерполирования функции по всей сетке, называются **квадратурными формулами интерполяционного типа**.

Имеем

$$\int_a^b \rho(x)f(x)dx \approx \sum_{k=0}^n c_k f(x_k); \quad c_k = \int_a^b \rho(x) \frac{\omega(x)}{(x - x_k)\omega'(x_k)} dx.$$

Поскольку интерполяционный многочлен определен единственным образом, то и весовые коэффициенты квадратурной формулы интерполяционного типа однозначно выражаются через x_k, ρ, h, a, b .

Пример 6.1. Пусть весовая функция $\rho(x) \equiv 1$, $[a, b] = [-1; 1]$, $x_0 = -1$, $x_1 = 0$, $x_2 = 1$. Интерполяция по трем точкам дает следующие значения весовых коэффициентов c_k :

$$c_0 = \int_{-1}^1 \frac{x(x-1)}{(-1)(-2)} dx = \frac{1}{2} \int_{-1}^1 x(x-1) dx = \frac{1}{2} \left(\frac{1}{3}x^3 - \frac{1}{2}x^2 \right) \Big|_{-1}^1 = \frac{1}{3};$$

$$c_1 = \int_{-1}^1 \frac{(x+1)(x-1)}{(+1)(-1)} dx = - \left(\frac{1}{3}x^3 - x \right) \Big|_{-1}^1 = \frac{4}{3};$$

$$c_2 = \int_{-1}^1 \frac{(x+1)x}{2 \cdot 1} dx = \frac{1}{2} \left(\frac{1}{3}x^3 + \frac{1}{2}x^2 \right) \Big|_{-1}^1 = \frac{1}{3}.$$

Отсюда

$$\int_{-1}^1 f(x) dx \approx \frac{1}{3} [f(-1) + 4f(0) + f(1)],$$

т. е. получена формула Симпсона при $h = 1$. •

Пример 6.2. В условиях предыдущего примера будем искать такую линейную функцию $\varphi(x) = ax + b$, что

$$\sum_{i=0}^2 [f(x_i) - \varphi(x_i)]^2 \rightarrow \min.$$

Опуская выкладки, ранее уже приводившиеся при описании *метода наименьших квадратов* (см. пример 4.3), получаем линейный полином наилучшего среднеквадратичного приближения

$$\varphi(x) = \frac{1}{3} [f(x_0) + f(x_1) + f(x_2)] + \frac{1}{2} [f(x_2) - f(x_0)] x.$$

Отсюда следует квадратурная формула неинтерполяционного типа

$$I = \int_{-1}^{+1} f(x) dx \approx \frac{2}{3} [f(x_0) + f(x_1) + f(x_2)]. \quad \bullet$$

Теорема 6.1. Квадратурная формула интерполяционного типа, построенная по $n + 1$ узлам x_0, x_1, \dots, x_n , является точной для любого полинома степени n . Для $n + 1$ раз непрерывно дифференцируемой на отрезке $[a, b]$ функции погрешность квадратурной формулы составляет

$$|\psi_h| \leq \frac{M_{n+1}}{(n+1)!} \int_a^b \rho(x) |\omega(x)| dx,$$

где $M_{n+1} = \|f^{(n+1)}\|_C$.

◀ Заменим на отрезке $[a, b]$ функцию $f(x)$ интерполяционным многочленом Лагранжа, тогда верно следующее соотношение:

$$f(x) = L_n(x) + r_n(x),$$

где $r_n(x) = \frac{f^{(n+1)}(\xi)}{(n+1)!} \omega(x)$.

Отсюда

$$|\psi_h| = \left| \int_a^b \rho(x) [f(x) - L_n(x)] dx \right| \leq \frac{M_{n+1}}{(n+1)!} \int_a^b \rho(x) |\omega(x)| dx.$$

Если $f(x)$ — полином, степень которого меньше или равна n , то $M_{n+1} = 0$ и, следовательно, $|\psi_h| = 0$, т. е. формула точна. ►

Справедлива и обратная теорема.

Теорема 6.2. Если квадратурная формула

$$I_h = \sum_{k=0}^n d_k f(x_k)$$

точна для любого полинома степени n , то она является квадратурной формулой интерполяционного типа.

◀ Необходимо доказать, что для произвольного k коэффициент $d_k = c_k$, где c_k — определенные выше весовые коэффициенты квадратурной формулы интерполяционного типа. Рассмотрим многочлен степени n

$$f(x) = \varphi_k(x) = \frac{\omega(x)}{(x - x_k)\omega'(x_k)},$$

удовлетворяющий условиям $\varphi_k(x_i) = \delta_{ki}$. Поскольку квадратурная формула точна для таких функций $f(x)$, то

$$\int_a^b \rho(x) f(x) dx = \int_a^b \rho(x) \varphi_k(x) dx = c_k = \sum_{i=0}^n \varphi_k(x_i) d_i = d_k.$$

Следовательно, $d_k = c_k$ для всех k . ►

Замечание 6.1. Формулы Симпсона, трапеций, прямоугольников — квадратурные формулы интерполяционного типа, при этом

значения функции интерполируются полиномами второго, первого и нулевого порядков соответственно.

Замечание 6.2. Формула центральных прямоугольников имеет повышенный порядок точности по сравнению с формулами левых или правых прямоугольников за счет выбора узла квадратурной формулы, расположенного симметрично относительно центра сеточного интервала. В общем случае условия симметричного выбора также позволяют повысить порядок точности.

Замечание 6.3. Оценку погрешности ψ_h можно улучшить: для формулы Симпсона ψ_h определяется в действительности четвертой производной, а не третьей $M_{n+1} = M_{2+1}$. Это происходит из-за учета симметрии в выводе оценки погрешности.

Определение. Квадратурные формулы интерполяционного типа на равномерной сетке $a = x_0 < x_1 < \dots < x_n = b$, где $x_k - x_{k-1} = h$, $k = \overline{1, n}$, называются **формулами Ньютона — Котеса**.

В случае равномерной сетки вид выражений для c_k может быть упрощен (упрощенные формулы приводить не будем).

Теорема 6.3. Квадратурная формула интерполяционного типа устойчива относительно возмущений f при условии знакопостоянства ее весовых коэффициентов.

◀ Пусть $I_h = \sum_{k=0}^n c_k f(x_k)$. Поскольку квадратурная формула точна при $f \equiv 1$, то

$$\sum_{k=0}^n c_k = \int_a^b \rho(x) dx = M.$$

Таким образом, в случае знакопостоянных c_k

$$\sum_{k=0}^n |c_k| = |M|$$

и не зависит от n .

Пусть

$$I_h + \delta I_h = \sum_{k=0}^n c_k [f(x_k) + \delta f(x_k)] = I_h + \sum_{k=0}^n c_k \delta f(x_k).$$

Тогда

$$|\delta I_h| \leq \sum_{k=0}^n |c_k| |\delta f(x_k)| \leq \|\delta f\|_C |M|.$$

Следовательно, квадратурная формула интерполяционного типа устойчива относительно возмущений f . ►

При рассмотрении интерполяции (см. 4.3.2) уже отмечалась неустойчивость глобальной интерполяции полиномами при больших n . Аналогично и при использовании ее в квадратурах с $n \geq 5$ должны проявляться эффекты неустойчивости. Например, известно, что при $n \geq 10$ и $\rho = 1$ появляются как положительные, так и отрицательные c_k . В результате $\sum_{k=0}^n |c_k|$ может превысить $|M|$ и квадратура будет неустойчивой. Во избежание подобных ошибок следует использовать, например, кусочно-полиномиальную аппроксимацию функций.

6.3. Квадратурные формулы Гаусса

Ранее узлы квадратурной формулы x_0, x_1, \dots, x_n (всего $n+1$ узлов) считались заданными (см. 6.1, 6.2). При этом была построена квадратурная формула, точная для полиномов степени, меньшей либо равной n , при выбранных весовых коэффициентах c_k , $k = \overline{0, n}$.

Построим квадратурную формулу на $n+1$ узлах так, чтобы она была точной для полиномов максимальной степени m . При этом будем выбирать весовые коэффициенты c_k и узлы x_k .

Запишем *условия точности квадратурной формулы* для полиномов степени не выше m :

$$\sum_{k=0}^n c_k x_k^\alpha = \int_a^b \rho x^\alpha dx, \quad \alpha = \overline{0, m}.$$

Имеем $2(n+1)$ неизвестных (c_k и x_k), следовательно, для их определения требуется $2(n+1)$ уравнений. Тогда максимальное $\alpha = m = 2n + 1$. Для нахождения c_k и x_k необходимо решить полученную систему.

Пример 6.3. Построим квадратурную формулу Гаусса для случая $\rho = 1$, $[a, b] = [-1; 1]$. Рассмотрим сетки, состоящие из одного ($n = 0$) и двух ($n = 1$) узлов.

1. Если $n = 0$, то максимальная степень полинома, для которого формула Гаусса точна, $m = 1$. Тогда система уравнений для поиска c_k и x_k примет вид

$$c_0 x_0^0 = \int_{-1}^1 x^0 dx = 2; \quad c_0 x_0^1 = \int_{-1}^1 x^1 dx = 0.$$

Следовательно, $c_0 = 2$, $x_0 = 0$ и квадратурная формула

$$\int_{-1}^{+1} f(x) dx \approx 2f(0)$$

точна для многочленов первой степени (формула центральных прямоугольников).

2. Если $n = 1$, то $m = 3$ и система уравнений примет вид

$$c_0 x_0^0 + c_1 x_1^0 = \int_{-1}^1 x^0 dx = 2;$$

$$c_0 x_0^1 + c_1 x_1^1 = \int_{-1}^1 x^1 dx = 0;$$

$$c_0x_0^2 + c_1x_1^2 = \int_{-1}^1 x^2 dx = \frac{2}{3};$$

$$c_0x_0^3 + c_1x_1^3 = \int_{-1}^1 x^3 dx = 0,$$

откуда

$$c_0 = c_1 = 1; \quad x_0 = -x_1 = -\frac{1}{\sqrt{3}},$$

и квадратурная формула

$$\int_{-1}^1 f(x) dx \approx f\left(-\frac{1}{\sqrt{3}}\right) + f\left(\frac{1}{\sqrt{3}}\right)$$

точна на полиномах, степень которых меньше или равна трем. •

Теорема 6.4. Квадратурная формула вида $I \approx I_h = \sum_{k=0}^n c_k f(x_k)$

точна на любых многочленах степени $m \leq 2n + 1$ тогда и только тогда, когда выполнены два условия:

1) $\int_a^b \rho(x)\omega(x)q(x) dx = 0$ для любого многочлена $q(x)$ степени,

меньшей $n + 1$, т. е. полином $\omega(x) = \prod_{k=0}^n (x - x_k)$ ортогонален $q(x)$ с весовой функцией $\rho(x)$;

2) на множестве узлов x_0, x_1, \dots, x_n эта формула является квадратурной формулой интерполяционного типа, т. е. весовые коэффициенты

$$c_k = \int_a^b \rho(x) \frac{\omega(x)}{(x - x_k)\omega'(x_k)} dx, \quad k = \overline{0, n}.$$

◀ Докажем необходимость приведенных условий. Формула точна для любого многочлена, степень которого не выше

$m = 2n + 1$. Значит, она точна и для $\omega(x)q(x)$, имеющего степень не выше m . Следовательно,

$$\int_a^b \rho(x)\omega(x)q(x)dx = \sum_{k=0}^n c_k \omega(x_k)q(x_k) = 0,$$

так как $\omega(x_k) = 0$ для любого k . Условие 2 выполняется в силу теоремы 6.2 (так как $m = 2n + 1 > n$, для которого теорема и доказана).

Докажем теперь достаточность приведенных условий. Пусть $f(x)$ — многочлен степени $m \leq 2n + 1$. Тогда, разделив его на $\omega(x)$, получим $f(x) = \omega(x)q(x) + r(x)$, где $q(x)$ и $r(x)$ — многочлены степени не выше n , так как $\omega(x)$ имеет степень $n + 1$. Вследствие того что полученное выражение есть квадратурная формула интерполяционного типа, она точна для $r(x)$:

$$\begin{aligned} \int_a^b \rho(x)r(x)dx &= \sum_{k=0}^n c_k [f(x_k) - \omega(x_k)q(x_k)] = \\ &= \sum_{k=0}^n c_k f(x_k) = \int_a^b \rho(x)[f(x) - \omega(x)q(x)]dx = \int_a^b \rho(x)f(x)dx. \end{aligned}$$

Таким образом, формула точна и для многочлена степени m . ►

Следствие 6.1. Уравнения для определения $n + 1$ узлов квадратурной формулы можно записать в виде

$$\int_a^b \rho(x)\omega(x)x^\alpha dx = 0, \quad \alpha = \overline{0, n}.$$

Это утверждение вытекает из условия 1 теоремы 6.4. Оно значительно упрощает процедуру нахождения узлов квадратурной формулы.

Полученные уравнения для определения узлов квадратурной формулы используются не только в теории методов численного

интегрирования. Эти же уравнения возникают при построении различных ортогональных систем полиномов (Чебышёва, Лагерра, Эрмита и др.) в задаче поиска их корней. Выбор весовой функции и области изменения аргумента приводит к той или иной системе ортогональных полиномов.

Теорема 6.4 не дает ответа на вопросы, сколько будет найдено различных решений полученной системы, т. е. наборов (x_0, x_1, \dots, x_n) , будут ли они лежать на отрезке $[a, b]$, будут ли они все различны и т. п. Не рассматривая эти вопросы, укажем лишь, что при $\rho(x) > 0$ такой многочлен $\omega(x)$ существует, единствен, все его корни различны и расположены на отрезке $[a, b]$.

Квадратурные формулы, удовлетворяющие приведенным выше условиям точности, называются **формулами Гаусса** или **квадратурными формулами наивысшей алгебраической степени точности**.

Легко видеть, что для многочлена степени $2(n+1)$ формула Гаусса, вообще говоря, не является точной. Действительно, пусть $f(x) = \omega^2(x)$ — многочлен степени $2(n+1)$,

$$I = \int_a^b \rho(x)f(x)dx > 0,$$

при этом

$$\sum_{k=0}^n c_k f(x_k) = \sum_{k=0}^n c_k \omega^2(x_k) = 0.$$

Поскольку квадратурная формула Гаусса точна вплоть до степени $m = 2n+1$, она точна и для многочлена степени $2n$

$$f(x) = \varphi_k^2(x) = \left(\frac{\omega(x)}{(x - x_k)\omega'(x_k)} \right)^2.$$

В силу того что

$$\int_a^b \rho \varphi_i^2 dx = \sum_{k=0}^n c_k \varphi_i^2(x_k) = \sum_{k=0}^n c_k \delta_{ik}^2 = c_i = \int_a^b \rho \varphi_i^2 dx > 0,$$

все коэффициенты $c_i > 0$. Следовательно, квадратурная формула Гаусса устойчива относительно возмущений функции $f(x)$ (см. теорему 6.3). Поэтому на практике формулы Гаусса применяют до $n \leq 100$.

В чем причина устойчивости формулы относительно погрешностей? Очевидно, что устойчивость есть результат неравномерности сетки, специально построенной для интегрируемой функции (см. 4.3.3).

Погрешность квадратурной формулы Гаусса можно представить в виде

$$\psi_h = \frac{1}{(2n+2)!} \int_a^b \rho(x) \omega^2(x) f^{(2n+2)}(\xi) dx, \quad \xi = \xi(x).$$

Для частных случаев весовых функций $\rho(x)$ узлы и весовые коэффициенты квадратурных формул Гаусса вычислены и приведены в справочниках.

Пример 6.4. Построим квадратурную формулу для вычисления интеграла $\int_{-1}^1 f(x) dx$, точную для полиномов второй степени.

Потребуем, чтобы число узлов квадратуры x_k , $k = \overline{0, n}$, было минимальным. Поскольку $m = 2n + 1$ нечетно, то удовлетворить требуемому условию точности при минимальном числе узлов можно лишь при $n = 1$, т. е. при $m = 3$.

Составим систему уравнений для определения весовых коэффициентов c_k и узлов x_k из условия точности для полиномов второй степени:

$$c_0x_0^0 + c_1x_1^0 = \int_{-1}^1 x^0 dx = 2;$$

$$c_0x_0^1 + c_1x_1^1 = \int_{-1}^1 x^1 dx = 0;$$

$$c_0x_0^2 + c_1x_1^2 = \int_{-1}^1 x^2 dx = \frac{2}{3}.$$

Получена система трех уравнений с четырьмя неизвестными.

Для того чтобы определить единственное решение задачи, необходимо ввести дополнительное условие для искомых величин. Оно может быть продиктовано дополнительной внешней информацией, например требованием симметрии формулы, т. е. условием $x_0 = -x_1$, которое приводит к квадратурной формуле

$$\int_{-1}^1 f(x) dx \approx f\left(-\frac{1}{\sqrt{3}}\right) + f\left(\frac{1}{\sqrt{3}}\right).$$

Эта формула получена ранее в примере 6.3, она будет точной и для полиномов третьей степени.

Выберем другое дополнительное условие: $x_1 = 1$, тогда квадратурная формула имеет вид

$$\int_{-1}^1 f(x) dx \approx \frac{3}{2} f\left(-\frac{1}{3}\right) + \frac{1}{2} f(1).$$

Данная формула точна для полиномов не выше второй степени.

Отметим, что если в качестве дополнительного условия выбрать $x_0 = 0$, то система, из которой определяются весовые коэффициенты квадратурной формулы, становится несовместной. •

Пример 6.5. Построим квадратурную формулу для вычисления интеграла $\int_a^b f(x) dx$.

Для построения такой формулы достаточно выполнить преобразование координат, переводящее отрезок $[-1; 1]$ в $[a, b]$, и воспользоваться уже известной квадратурной формулой Гаусса для отрезка $[-1; 1]$. Примеры таких формул приведены в табл. 6.1.

Таблица 6.1

Узлы и веса n -точечной квадратурной формулы Гаусса, точной для полиномов степени не выше m , на отрезке $[-1; 1]$ с весом $\rho(x) = 1$

n	m	x_k	c_k	n	m	x_k	c_k
1	1	0	2	4	7	$\pm \sqrt{\frac{3}{7} + \frac{2}{7}\sqrt{\frac{6}{5}}}$	$\frac{18 - \sqrt{30}}{36}$
2	3	$\pm 1/\sqrt{3}$	1			$\pm \sqrt{\frac{3}{7} - \frac{2}{7}\sqrt{\frac{6}{5}}}$	$\frac{18 + \sqrt{30}}{36}$
3	5	$\pm \sqrt{3/5}$	5/9				
		0	8/9				

Используем линейное преобразование координат:

$$x = \frac{a+b}{2} + \frac{b-a}{2}t,$$

где $t \in [-1; 1]$. Тогда

$$\begin{aligned} \int_a^b f(x) dx &= \frac{b-a}{2} \int_{-1}^1 f\left(\frac{a+b}{2} + \frac{b-a}{2}t\right) dt \approx \\ &\approx \sum_{k=0}^n \frac{b-a}{2} c_k f\left(\frac{a+b}{2} + \frac{b-a}{2}t_k\right). \end{aligned}$$

Указанный прием можно распространить на любые квадратурные формулы. •

6.4. Интегрирование быстроосциллирующих функций

Пусть требуется вычислить интеграл

$$I = \int_a^b f(x) e^{i\nu x} dx$$

с большим значением ν (здесь i — мнимая единица: $i^2 = -1$). При использовании стандартного подхода каждый полупериод π/ν необходимо разбить хотя бы на 10 отрезков. Тогда на интервале (a, b) нужно иметь следующее число узлов квадратурной формулы:

$$N \approx 10 \frac{b-a}{\pi} \nu.$$

При больших ν число узлов N также будет большим, что может существенно затруднить численное интегрирование в силу неустойчивости формул интерполяционного типа или сложности непосредственного построения квадратурных формул Гаусса. Поставленную задачу можно решить, если считать экспоненту под знаком интеграла весовой функцией, т. е. принять, что $\rho(x) = e^{i\nu x}$, и использовать квадратурные формулы типа Ньютона — Котеса. Так, если $L_n(x)$ — интерполяционный полином Лагранжа, то

$$I \approx \sum_{k=0}^n c_k f(x_k), \quad c_k = \int_a^b e^{i\nu x} \frac{\omega(x)}{(x-x_k)\omega'(x_k)} dx, \quad k = \overline{0, n}.$$

При этом погрешность

$$\psi_h = \int_a^b e^{i\nu x} [f(x) - L_n(x)] dx.$$

Получаемые квадратурные формулы носят названия **квадратурных формул Филона**.

При интегрировании быстроосциллирующих функций можно использовать не только глобальную, но и локальную интерполяцию, например сплайны различного порядка. Такие формулы также называются формулами Филона.

Рассмотрим вариант замены подынтегральной функции $f(x)$ на ее среднее значение $f(x_{i-1/2})$, где

$$x_{i-1/2} = x_{i-1} + \frac{h}{2} = \frac{x_i + x_{i-1}}{2}$$

на равномерной (для простоты) сетке. Тогда получим квадратурную формулу Филона типа формулы центральных прямоугольников:

$$I_{h,i} = f(x_{i-1/2}) \int_{x_{i-1}}^{x_i} e^{\tilde{i}\nu x} dx = \frac{2}{\nu} f(x_{i-1/2}) e^{\tilde{i}\nu x_{i-1/2}} \sin \frac{\nu h}{2}.$$

Погрешность построенной квадратурной формулы можно оценить по аналогии с тем, как это было сделано в 6.1.2.

Ясно, что при необходимости можно построить формулы на основе линейных локальных сплайнов или сплайнов более высокого порядка.

6.5. Вычисление несобственных интегралов I и II рода

Пусть требуется вычислить несобственный интеграл, т. е. интеграл I рода по неограниченному участку или интеграл II рода от неограниченной функции.

Интеграл I рода. Рассмотрим вычисление интеграла

$$I = \int_a^{\infty} f(x) dx,$$

где $f(x)$ — гладкая функция. Существует ряд приемов расчета.

1. Замена переменной для получения интеграла по конечному участку.

Пример 6.6. При вычислении интеграла $I = \int_a^{\infty} f(x) dx$ можно выполнить следующую замену переменной x :

$$x = \frac{a}{1-t}: (a, \infty) \rightarrow (0; 1); \quad dx = \frac{a}{(1-t)^2} dt.$$

Тогда

$$I = \int_0^1 \frac{a}{(1-t)^2} f\left(\frac{a}{1-t}\right) dt. \quad \bullet$$

Если будет получена ограниченная результирующая подынтегральная функция, то далее для расчета применяют обычные квадратурные формулы.

2. Обрезание верхнего предела. По определению несобственного интеграла I рода

$$I = \int_a^{\infty} f(x) dx = \lim_{A \rightarrow \infty} \int_a^A f(x) dx = \lim_{A \rightarrow \infty} I_A.$$

Вычисление I_A можно проводить обычным способом. Иногда, заменив точное значение I интеграла I рода приближенным значением I_A , удается оценить погрешность $\int_A^{\infty} f(x) dx$, что позволяет выбрать предел A , исходя из ее требуемого значения. Чаще всего вычисляют пару значений I_{A_1}, I_{A_2} . Если $A_2 > A_1$ (существенно), а I_{A_1}, I_{A_2} различаются менее, чем на заданную малую величину, то процесс прекращают.

Однако этот алгоритм расчета несобственного интеграла весьма ненадежен при его медленной сходимости, например если подынтегральная функция меняет знак.

3. Использование квадратурной формулы Гаусса. Для интеграла

$$I = \int_a^{\infty} \rho(x) f(x) dx$$

и для соответствующей весовой функции $\rho(x)$ выбирают набор узлов $\{x_k\}$. Далее применяют квадратурную формулу Гаусса.

Пример 6.7. Пусть необходимо вычислить значения функции Эйри

$$E_i(x) = \int_x^{\infty} \frac{e^{-t}}{t} dt = \int_0^{\infty} \frac{e^{-t-x}}{t+x} dt = e^{-x} \int_0^{\infty} \frac{e^{-t}}{t+x} dx.$$

В этом случае положим $\rho(t) = e^{-t}$. Соответствующие узлы квадратурной формулы являются нулями полинома Лагерра $\{x_k\}$. Следовательно, получим некоторую аппроксимирующую формулу

$$E_i(x) \approx e^{-x} \sum_{k=0}^n \frac{\gamma_k}{x_k + x},$$

где $\gamma_k = c_k$ — весовой коэффициент квадратурной формулы. •

4. Использование нестандартных формул. Необходимо приблизить $f(x)$ какой-то функцией, интеграл от которой легко вычисляется. Например, если приблизить $f(x)$ экспонентой на отрезке $[x_n - h/2, +\infty)$, потребовав точной передачи производной и функции в точке x_n , то получим

$$f(x) \approx \alpha e^{-\beta x}; \quad f'(x_n) = -f(x_n) e^{\beta x_n - \beta x_n} \beta,$$

откуда

$$\alpha = f(x_n) e^{\beta x_n}; \quad \beta = -f'(x_n)/f(x_n);$$

$$\int_{x_n - h/2}^{+\infty} f(x) dx \approx \frac{\alpha}{\beta} e^{-\beta(x_n - h/2)}.$$

Для вычисления интеграла по отрезку $[a, x_n - h/2]$ используют обычные квадратурные формулы.

Интеграл II рода. Пусть требуется вычислить интеграл

$$I = \int_a^b f(x) dx,$$

где $f(x)$ не ограничена в точке a . Существуют разнообразные приемы вычислений.

1. **Аддитивное выделение особенности.** Предположим, что подынтегральную функцию можно представить в виде суммы $f(x) = \varphi(x) + \psi(x)$ так, что $\varphi(x)$ содержит особенность (является неограниченной) и интегрируется аналитически, а $\psi(x)$ не содержит особенности и интегрируется численно.

Пример 6.8. Применим аддитивное выделение особенности

для вычисления интеграла $\int_0^1 \frac{f(x)}{\sqrt{x}} dx$. Воспользовавшись разложением функции $f(x)$ по формуле Тейлора, получим

$$\begin{aligned} \int_0^1 \frac{f(x)}{\sqrt{x}} dx &= \int_0^1 \frac{1}{\sqrt{x}} [f(x) - f(0) - xf'(0)] dx + \\ &\quad + \int_0^1 \frac{f(0)}{\sqrt{x}} dx + \int_0^1 f'(0) \sqrt{x} dx. \end{aligned}$$

Два последних интеграла вычисляют аналитически. В первом интеграле выделено столько слагаемых, чтобы можно было использовать квадратурную формулу необходимой точности. •

2. **Мультипликативное выделение особенности.** Предположим, что $f(x) = \varphi(x)\rho(x)$, где $\varphi(x)$ — гладкая и ограниченная функция, а $\rho(x) > 0$, интегрируема, но содержит особенность.

В этом случае можно использовать квадратурную формулу Гаусса для вычисления интеграла, считая $\rho(x)$ весовой функцией.

Пример 6.9. Используем мультипликативное выделение особенности для вычисления интеграла $\int_{-1}^1 \frac{e^x}{\sqrt{1-x^2}} dx$. Возьмем в качестве весовой функции $\rho(x) = \frac{1}{\sqrt{1-x^2}}$, а в качестве узлов квадратурной формулы нули многочленов Чебышёва:

$$x_i = \cos \frac{\pi}{n} \left(i - \frac{1}{2} \right), \quad i = \overline{1, n}.$$

Тогда получим

$$\int_{-1}^1 \frac{e^x}{\sqrt{1-x^2}} dx \approx \frac{\pi}{n} \sum_{i=1}^n e^{x_i}. \quad \bullet$$

3. Использование нестандартных аппроксимаций. Пусть $f(x) = \rho(x)\varphi(x)$, где функция $\varphi(x)$ изменяется слабо по сравнению с функцией $\rho(x)$. Тогда на отрезке $[x_{i-1}, x_i]$ функция $f(x) \approx \rho(x)\varphi(x_{i-1/2})$ и интеграл

$$\int_{x_{i-1}}^{x_i} f(x) dx \approx \varphi(x_{i-1/2}) \int_{x_{i-1}}^{x_i} \rho(x) dx.$$

Часто последний интеграл можно вычислить аналитически, как это было сделано, например, в 6.4. Напомним, что один из методов вычисления интегралов от быстроосциллирующих функций — это использование указанных аппроксимаций.

4. Обрезание нижнего предела. По определению несобственного интеграла II рода

$$I = \int_a^b f(x) dx = \lim_{c \rightarrow a+0} \int_c^b f(x) dx = \lim_{c \rightarrow a+0} I_c.$$

Далее алгоритм вычисления аналогичен алгоритму обрезания верхнего предела для интегралов I рода.

6.6. Вычисление кратных интегралов

Пусть необходимо вычислить интеграл

$$I = \iint_G f(x, y) dx dy.$$

Для этого можно использовать метод ячеек, последовательное интегрирование, конечно-элементный подход и др.

Метод ячеек. Пусть область G представляет собой прямоугольник: $G = [a, b] \times [c, d]$. Разобьем отрезки $[a, b]$, $[c, d]$ на M и N частей соответственно. В каждом элементарном прямоугольнике выберем центральную точку и запишем формулу для приближенного вычисления интеграла в виде

$$I_h = \sum_{i=1}^M \sum_{j=1}^N f(x_{i-1/2}, y_{j-1/2}) h_{x,i} h_{y,j},$$

где $h_{x,i} = x_i - x_{i-1}$; $h_{y,j} = y_j - y_{j-1}$.

Как и в одномерном случае (см. 6.1.2), получим оценку погрешности $\psi_h = O(h_x^2 + h_y^2)$, поскольку $f(x, y)$ вычисляется в центре прямоугольника.

Значение погрешности определяется вторыми производными f''_{x^2} , f''_{y^2} , f''_{xy} . Ее оценка может быть получена практически так же, как и соответствующая оценка для квадратурной формулы центральных прямоугольников. Для этого достаточно использовать разложение функции $f(x)$ под знаком интеграла по формуле Тейлора для случая двух независимых переменных.

В случае если элементарная ячейка не является прямоугольником, тот же порядок точности будет получен при вычислении $f(x, y)$ в точке, совпадающей с центром тяжести данной ячейки.

Координаты (\bar{x}, \bar{y}) центра тяжести ячейки G_h определяются выражениями

$$\bar{x} = \frac{1}{S_h} \iint_{G_h} x \, dx \, dy; \quad \bar{y} = \frac{1}{S_h} \iint_{G_h} y \, dx \, dy,$$

где $S_h = \iint_{G_h} dx \, dy$.

При получении оценки погрешности интегралы от линейных слагаемых в формуле Тейлора оказываются равными нулю. В результате погрешность остается той же.

Последовательное интегрирование. Пусть область интегрирования G можно представить в виде

$$\begin{aligned} G &= \{x \in [a, b] : \varphi_1(x) \leq y \leq \varphi_2(x)\} = \\ &= \{y \in [c, d] : \psi_1(y) \leq x \leq \psi_2(y)\}. \end{aligned}$$

Тогда двойной интеграл можно вычислить с помощью последовательного интегрирования. Представим интеграл I в виде

$$I = \iint_G f(x, y) \, dx \, dy = \int_a^b F(x) \, dx,$$

где $F(x) = \int_{\varphi_1(x)}^{\varphi_2(x)} f(x, y) \, dy$, или в виде

$$I = \iint_G f(x, y) \, dx \, dy = \int_c^d R(y) \, dy,$$

где $R(y) = \int_{\psi_1(y)}^{\psi_2(y)} f(x, y) \, dx$.

В этом случае можно выполнить численное интегрирование по y для фиксированного набора x , а далее аналогично по x .

Последовательное применение квадратурных формул по обоим направлениям приводит к кубатурным формулам, которые получают прямым произведением одномерных квадратурных формул. При этом можно использовать все квадратурные формулы, полученные ранее.

Конечно-элементный подход. Если приближенно представить подынтегральную функцию в виде

$$f(x, y) \approx \sum_{i=1}^N f(x_i, y_i) \varphi_i(x, y),$$

где $\varphi_i(x, y)$ — двумерные базисные функции конечных элементов, то интеграл I может быть вычислен по квадратурной формуле

$$I \approx I_h = \sum_{i=1}^N f(x_i, y_i) \int_G \varphi_i(x, y) dx dy.$$

При этом используется двумерная интерполяция, например, на треугольных сетках.

6.7. Численное дифференцирование

Пусть известны $n + 1$ значений функции $f(x)$ в $n + 1$ точках x_0, x_1, \dots, x_n . Необходимо вычислить производную $f'(x)$. Для этого можно использовать сплайн-интерполяцию, глобальную полиномиальную интерполяцию, а также разностные соотношения.

Сплайн-интерполяция. Как известно, для функции $f(x) \in C^{(4)}(a, b)$ (при условии точного задания $f''(a)$ и $f''(b)$) ее кубическая сплайн-интерполяция $\varphi(x)$ такова, что

$$\|\varphi' - f'\| \leq C_1 M_4 h^3; \quad \|\varphi'' - f''\| \leq C_2 M_4 h^2.$$

Таким образом, сплайн позволяет получить f' , f'' путем прямого дифференцирования и с высокой точностью. Здесь $M_4 = \|f^{(4)}\|_C$.

Глобальная полиномиальная интерполяция. Рассмотрим интерполяционный полином в форме Ньютона:

$$\begin{aligned} L_n(x) = & f(x_0) + (x - x_0)f(x_0, x_1) + \\ & + (x - x_0)(x - x_1)f(x_0, x_1, x_2) + \dots \\ & \dots + (x - x_0)(x - x_1) \cdots (x - x_{n-1})f(x_0, x_1, \dots, x_n). \end{aligned}$$

Теорема 6.5. Пусть $f(x) \in C^{(n)}[a, b]$. Тогда

$$\exists \xi \in [a, b]: f(x_0, \dots, x_n) = \frac{f^{(n)}(\xi)}{n!}.$$

◀ Рассмотрим погрешность интерполирования $r_n(x) = f(x) - L_n(x)$. Эта функция имеет $n+1$ нулей на отрезке $[a, b]$. Соответственно, ее производная r'_n имеет n нулей, расположенных между нулями r_n , поскольку указанные нули некратные, вторая производная имеет $n-1$ нулей и т. д. Следовательно, n -я производная имеет хотя бы один нуль $\xi \in [a, b]$:

$$\begin{aligned} r_n^{(n)}(\xi) = 0 &= f^{(n)}(\xi) - n!f(x_0, \dots, x_n), \\ \text{откуда } f(x_0, \dots, x_n) &= \frac{f^{(n)}(\xi)}{n!}. \end{aligned} \quad \blacktriangleright$$

Согласно теореме 6.5, первая производная функции может быть приближенно вычислена с помощью формулы

$$f(x_0, x_1) = \frac{f(x_1) - f(x_0)}{x_1 - x_0},$$

а вторая производная — с помощью формулы

$$2!f(x_0, x_1, x_2) = \frac{2!}{x_2 - x_0} \left(\frac{f(x_2) - f(x_1)}{x_2 - x_1} - \frac{f(x_1) - f(x_0)}{x_1 - x_0} \right)$$

и т. д. Однако при этом неизвестно, к каким точкам относятся значения производных. Теорема 6.5 указывает лишь на то, что $\xi \in [a, b]$, где a, b — границы отрезка интерполирования.

Для вычисления $y' = f'$ также можно использовать интерполянт L_n . Очевидно, что для вычисления $f^{(n)}$ необходимо не менее $n+1$ точек интерполяции.

Теорема 6.6. Пусть $f(x) \in C^{(n+1)}[a, b]$, $L_n(x)$ — интерполяционный полином, построенный по значениям функции в точках сетки $\Omega_h = \{a = x_0 < x_1 < x_2 < \dots < x_n = b\}$. Тогда погрешность вычисления производной q -го порядка ($q = \overline{1, n}$) удовлетворяет оценке

$$\|f^{(q)} - L_n^{(q)}\|_C \leq \frac{1}{(n+1-q)!} (b-a)^{n+1-q} \|f^{(n+1)}\|_C.$$

◀ Рассмотрим остаток $r_n(x) = f(x) - L_n(x)$. Эта функция имеет $n+1$ нулей на отрезке $[a, b]$. Соответственно, ее производная r'_n имеет n нулей, расположенных между нулями r_n , поскольку указанные нули некратные, вторая производная имеет $n-1$ нулей и т. д. Следовательно, q -я производная функции $r_n(x)$ имеет не менее $n+1-q$ нулей на отрезке $[a, b]$.

Таким образом, значения $f^{(q)}(x)$ и $L_n^{(q)}(x)$ совпадают по крайней мере в $n+1-q$ точках отрезка $[a, b]$, т. е. $L_n^{(q)}(x)$ есть интерполяционный полином для функции $f^{(q)}(x)$. Последняя имеет производную $(n+1-q)$ -го порядка, что позволяет стандартным образом оценить остаточный член интерполяции. При этом используется функция

$$\tilde{\omega}(x) = \prod_{i=0}^{n-q} (x - \tilde{x}_i),$$

где точки $a \leq \tilde{x}_0 < \dots < \tilde{x}_{n-q} \leq b$ — узлы интерполяции функции $f^{(q)}(x)$. В силу неопределенности положения точек интерполяции полученный остаток необходимо оценивать с помощью максимума $|\tilde{\omega}|$ по всем возможным расположениям точек. ►

Следствие 6.2. Рассмотрим случай равномерной сетки, на которой $b-a = hn$, где h — шаг сетки. Тогда при выполнении условий теоремы 6.6 оценка погрешности численного дифференцирования принимает вид

$$\|f^{(q)} - L_n^{(q)}\|_C \leq \frac{h^{n+1-q}}{(n+1-q)!} n^{n+1-q} \|f^{(n+1)}\|_C.$$

Полученная оценка бесполезна в предельном случае $q = n$, поскольку значения $f^{(n)}$ и $L_n^{(n)}$ совпадают лишь в одной точке, а неравенство содержит норму разности этих производных, взятых по всему отрезку интерполяции. Кроме того, правая часть неравенства не убывает при возрастании n .

Рассмотрим другой предельный случай, когда число $n + 1 - q$ велико. Здесь применение формулы Стирлинга дает оценку

$$\|f^{(q)} - L_n^{(q)}\|_C \leq \frac{h^{n+1-q}}{\sqrt{2\pi(n+1-q)}} e^n \|f^{(n+1)}\|_C.$$

Пример 6.10. Рассмотрим квадратичный интерполянт, построенный по заданным в трех точках x_{i-1} , x_i , x_{i+1} значениям функции f_{i-1} , f_i , f_{i+1} .

Интерполяция по трем точкам дает полином

$$\begin{aligned} L_2(x) = & f_{i-1} \frac{(x - x_i)(x - x_{i+1})}{(x_{i-1} - x_i)(x_{i-1} - x_{i+1})} + \\ & + f_i \frac{(x - x_{i-1})(x - x_{i+1})}{(x_i - x_{i-1})(x_i - x_{i+1})} + \\ & + f_{i+1} \frac{(x - x_{i-1})(x - x_i)}{(x_{i+1} - x_{i-1})(x_{i+1} - x_i)}. \end{aligned}$$

Его производная представляет собой линейную функцию

$$\begin{aligned} L'_2(x) = & f_{i-1} \frac{(2x - x_i - x_{i+1})}{(x_{i-1} - x_i)(x_{i-1} - x_{i+1})} + \\ & + f_i \frac{(2x - x_{i-1} - x_{i+1})}{(x_i - x_{i-1})(x_i - x_{i+1})} + \\ & + f_{i+1} \frac{(2x - x_{i-1} - x_i)}{(x_{i+1} - x_{i-1})(x_{i+1} - x_i)}. \end{aligned}$$

Из доказательства теоремы 6.6 следует, что значения $L'_2(x)$ на отрезке $[x_{i-1}, x_{i+1}]$ хотя бы дважды совпадут с точными значениями производной $f'(x)$. Точки этих совпадений заранее не известны. Тем не менее они лежат на отрезках $[x_{i-1}, x_i]$

и $[x_i, x_{i+1}]$. Отсюда понятно, что значения $L'_2(x_{i-1})$, $L'_2(x_i)$, $L'_2(x_{i+1})$ должны с некоторой погрешностью аппроксимировать точные значения производной исходной функции в этих точках. •

Разностные соотношения. Пусть $f(x_i)$ — значения функции $y = f(x)$ в точках сетки, для простоты равномерной:

$$x_i = a + ih, \quad i = \overline{0, n}; \quad h = \frac{b - a}{n}.$$

Необходимо вычислить приближенные значения производных в точках $x = x_i$.

Возможны следующие варианты:

- 1) $y_{\bar{x},i} = (y_i - y_{i-1})/h$ — **разностная производная назад**, или **левая разностная производная**;
- 2) $y_{x,i} = (y_{i+1} - y_i)/h$ — **разностная производная вперед**, или **правая разностная производная**;
- 3) $y_{\circ,x,i} = (y_{i+1} - y_{i-1})/(2h)$ — **центральная разностная производная**.

Найдем погрешность приведенных разностных соотношений. Пусть $y_i = f(x_i)$ — значение трижды непрерывно дифференцируемой функции $f(x)$ в точке x_i . Тогда

$$\begin{aligned} f(x_{i-1}) &= y_{i-1} = f(x_i) + \frac{1}{1!}f'(x_i)(x_{i-1} - x_i) + \\ &\quad + \frac{1}{2!}f''(x_i)(x_{i-1} - x_i)^2 + \frac{1}{3!}f'''(\xi_i)(x_{i-1} - x_i)^3; \\ f(x_{i+1}) &= y_{i+1} = f(x_i) + hf'(x_i) + \frac{1}{2}h^2f''(x_i) + \frac{1}{6}h^3f'''(\tilde{\xi}_i). \end{aligned}$$

Следовательно,

$$\begin{aligned} y_{\bar{x},i} &= f'(x_i) - \frac{1}{2}hf''(\xi_i^1); \quad y_{x,i} = f'(x_i) + \frac{1}{2}hf''(\xi_i^2); \\ y_{\circ,x,i} &= f'(x_i) + \frac{1}{3}h^2f'''(\xi_i^3). \end{aligned}$$

Отметим, что $y_{\bar{x},i} = y_{x,i-1}$, $y_{\bar{x},i+1} = y_{x,i}$, а центральная разностная производная $y_{\circ,x,i} = (y_{x,i} + y_{\bar{x},i})/2$ имеет повышенный порядок

аппроксимации. Это соответствует и разным знакам в погрешностях $y_{\bar{x},i}$, $y_{x,i}$. При этом левая и правая разностные производные есть производные линейного сплайна в соответствующих точках, а центральная разностная производная — производная квадратичного интерполянта в центре отрезка интерполяции на равномерной сетке.

Пример 6.11. Рассмотрим *метод Рунге — Ромберга* повышения точности формул численного дифференцирования на примере левой разностной производной. Для этого выберем точки x_{i-1}, x_i, x_{i+1} с заданными в них значениями функции. Вычислим левую разностную производную в точке x_{i+1} дважды для сеток с шагами h и $2h$.

Воспользуемся выражением для погрешности левой разностной производной и запишем

$$\begin{aligned} y_{\bar{x},i+1}^{(1)} &= \frac{1}{h}(y_{i+1} - y_i) = f'(x_{i+1}) + Ch + O(h^2); \\ y_{\bar{x},i+1}^{(2)} &= \frac{1}{2h}(y_{i+1} - y_{i-1}) = f'(x_{i+1}) + C \cdot 2h + O(h^2). \end{aligned}$$

Легко видеть, что главный член погрешности численного дифференцирования может быть исключен путем элементарных преобразований:

$$\tilde{y}_{\bar{x},i+1} = \frac{1}{2h}(3y_{i+1} - 4y_i + y_{i-1}) = f'(x_{i+1}) + O(h^2).$$

Это выражение представляет собой одностороннюю (левую) разностную производную второго порядка аппроксимации на трех точках. Точно такую же формулу можно получить при вычислении производной с использованием квадратичного интерполянта на равномерной сетке. Отметим, что полученная формула дает значение производной на границе рассматриваемого отрезка $[x_{i-1}, x_{i+1}]$, а не в его центре. Отсюда и следует отличие данной производной от центральной разностной производной.

Такой прием использован и при получении квадратурной формулы Симпсона (см. 6.1.4). Этот прием имеет общий характер и применим всегда при наличии как минимум двух выражений (соответствующих разным шагам сетки) с известной асимптотикой погрешности по некоторому малому параметру (в данном случае — шагу сетки). Не составляет труда записать его формулы при наличии главной части $O(h^p)$ остаточного члена погрешности. •

При использовании метода Рунге — Ромберга повышение точности происходит за счет удаления главного слагаемого погрешности некоторого численного выражения (Ch в примере 6.11). Для этого записывают систему двух уравнений, в которой одной неизвестной является главное слагаемое погрешности, а другой — уточняемая величина (левая разностная производная в примере 6.11). Далее главное слагаемое погрешности из системы уравнений исключают, получая более точное значение рассматриваемого выражения. Если бы главная часть погрешности равнялась Ch^p с известным p , то алгоритм был бы практически такой же.

Пример 6.12. Рассмотрим случай, в котором асимптотика главного члена погрешности некоторой численной формулы неизвестна, т. е. неизвестны параметры C и p . Формулы, связывающие точное и приближенные значения вычисляемой величины, содержат три неизвестные, поэтому для их определения необходимы три уравнения. Пусть в нашем распоряжении есть численные значения f_1, f_2, f_3 — приближения некоторой величины f — на сетках с шагами h, qh, q^2h соответственно. Тогда имеем систему трех уравнений

$$\begin{aligned}f_1 &= f + Ch^p + O(h^{p+1}); \\f_2 &= f + Cq^ph^p + O(h^{p+1}); \\f_3 &= f + Cq^{2p}h^p + O(h^{p+1}).\end{aligned}$$

Решая ее с погрешностью $O(h^{p+1})$, получим

$$f = f_1 + \frac{(f_1 - f_2)^2}{2f_2 - f_1 - f_3} + O(h^{p+1}), \quad p \approx \frac{1}{\ln q} \ln \frac{f_3 - f_2}{f_2 - f_1},$$

т. е. не только уточненное значение вычисляемой величины, но и эффективный порядок точности. •

Описанный в примере 6.12 алгоритм называется ***процессом Эйткена***.

Введем аппроксимацию ***второй разностной производной***:

$$\begin{aligned} y_{\bar{x}x} &= \frac{1}{h}(y_{\bar{x},i+1} - y_{\bar{x},i}) = \frac{1}{h}(y_{x,i} - y_{x,i-1}) = \\ &= \frac{1}{h^2}(y_{i+1} - 2y_i + y_{i-1}) = f''(x_i) + \frac{h^2}{12}f^{(4)}(\xi) \end{aligned}$$

в случае наличия непрерывной четвертой производной.

Можно ввести аппроксимацию третьей производной, а также производных более высокого порядка.

Отметим ***некорректность*** как ***численного дифференцирования***, так и вычисления производной функции непрерывного аргумента. В случае разностных производных операция вычитания и деления на малое число h ведет к существенному возрастанию погрешности. Например, если f известны с точностью Δf , то

$$\Delta f' = \frac{2\Delta f}{h}; \quad \Delta f'' = \frac{4\Delta f}{h^2}$$

и т. д. Полученная погрешность тем существеннее, чем меньше погрешность аппроксимации приближенной формулы.

Последние соотношения показывают, что формально правильное с точки зрения математического анализа устремление шага сетки к нулю для получения более точного результата численного дифференцирования может в случае приближенно заданной функции дать крайне неудовлетворительный результат.

Отсюда следует, что при численном дифференцировании необходимо принимать специальные меры для того, чтобы получать надежные результаты. Обычно подобные меры называют *регуляризацией дифференцирования*. Чаще всего они сводятся к поиску решения исходной задачи на некотором подпространстве исходного пространства, в котором содержатся более гладкие функции. Ранее термин «регуляризация» встречался при решении плохо обусловленных СЛАУ. Более подробно этот алгоритм может быть исследован при решении интегральных уравнений первого рода. Эти уравнения имеют прямое отношение к вычислению производной, так как процедура дифференцирования может быть сведена к решению интегрального уравнения Вольтерра первого рода.

Простейший пример регуляризации — выбор шага сетки, согласованного с точностью задания функции. Если, например, $|f''| \leq M_2$, то погрешность приближенного вычисления производной слабо скажется на результате при условии, что

$$\frac{2\Delta f}{h} \approx \frac{1}{2}hM_2,$$

т. е.

$$\Delta f \approx \frac{1}{4}h^2M_2.$$

В этом случае погрешность численного дифференцирования есть величина порядка погрешности аппроксимации. То же выражение можно использовать для определения длины шага сетки:

$$h \approx h_0 = 2 \sqrt{\frac{\Delta f}{M_2}}.$$

Легко видеть, что приведенная длина шага сетки оптимальна, т. е. обеспечивает минимум суммы $\frac{2\Delta f}{h} + h\frac{M_2}{2}$.

При вычислении старших производных выбрать оптимальный шаг еще сложнее.

Параметры сетки должны быть согласованы с решением задачи. На участках быстрого изменения решения (и, соответственно, больших значений производных) сетка должна быть мелкой и наоборот. Отсюда вытекает необходимость использования неравномерных сеток, которые позволяют получить более точное решение задачи с меньшими затратами ресурсов.

Некорректность дифференцирования имеет место и в дифференциальном случае. Рассмотрим функцию

$$f(x) = \varepsilon \sin\left(\frac{x}{\varepsilon^2}\right); \quad \|f\|_C = \varepsilon.$$

Ее производная

$$f'(x) = \frac{1}{\varepsilon} \cos\left(\frac{x}{\varepsilon^2}\right); \quad \|f'\|_C = \frac{1}{\varepsilon}, \quad \varepsilon \neq 0.$$

При $\varepsilon = 0$ функция $f(x)$ есть тождественный нуль, так что и ее производная равна нулю. В то же время, насколько бы малым ни было $\varepsilon \neq 0$ (насколько бы малой, но не нулевой, ни была норма $\|f\|_C$), норма производной окажется отличной от нуля, причем тем больше, чем меньше $\varepsilon = \|f\|_C$.

Вопросы и задания

1. Дайте определение квадратурной формулы. Что такое узлы и весовая функция квадратурной формулы?
2. Запишите квадратурную формулу прямоугольников и ее варианты. Как получить оценку погрешности этих формул?
3. Запишите квадратурную формулу трапеций. Как получить оценку погрешности этой формулы?
4. Запишите квадратурную формулу Симпсона. Как получить оценку погрешности этой формулы?

5. Какие квадратурные формулы называются формулами интерполяционного типа?
6. Какие квадратурные формулы называются формулами Ньютона — Котеса?
7. Приведите оценку погрешности квадратурных формул интерполяционного типа.
8. Дайте определение квадратурных формул Гаусса. Приведите оценку погрешности квадратурных формул Гаусса.
9. Приведите пример квадратурной формулы для интегрирования быстроосциллирующих функций.
10. Приведите примеры квадратурных формул для вычисления несобственных интегралов: а) I рода; б) II рода.
11. Приведите примеры квадратурных формул для вычисления кратных интегралов.
12. Приведите примеры методов численного дифференцирования.
13. Обоснуйте некорректность операции численного дифференцирования.
14. Каковы способы регуляризации операции численного дифференцирования?

Библиографические комментарии

Задачи вычисления определенных интегралов и производных функций — непременная составляющая любого курса математического анализа. Практически любой курс, в котором изучают методы вычислений, содержит материал, посвященный численному интегрированию. То же самое можно сказать и о дифференцировании. Поэтому задачи численного интегрирования и дифференцирования можно найти во всех книгах учебного характера, указанных в библиографии.

Отметим монографии [59] и [70], в которых рассмотрено построение квадратурных формул для различных классов функций, обладающих какими-либо особыми свойствами типа оптимальности. В указанных монографиях отмечается очевидная связь численных методов и теории функций. Справочный материал по численному интегрированию и дифференцированию содержится также в [1] и [32].

В работе [35] приведен обширный материал по регуляризации численного дифференцирования, включая чисто практические приемы, а в [36] описано применение так называемых квазиравномерных сеток, в частности для расчета интегралов и производных.

Метод регуляризации подробно представлен в работах [57, 67, 74, 75].

Материал по классическим ортогональным полиномам и их применению в вычислительной математике можно найти в справочнике [1] и монографии [58].

В работе [20] приведена оценка погрешности численного дифференцирования в случае функции, значительно менее гладкой, чем рассмотренная в настоящем пособии.

Один из важных методов решения задач вычислительной математики, в том числе задачи вычисления интегралов, — метод Монте-Карло — рассмотрен в работе [71]. Этот метод особенно эффективен при расчете многомерных интегралов.

В монографии [46] описаны и многие другие методы численного интегрирования, полезные при решении больших задач.

7. ЧИСЛЕННОЕ РЕШЕНИЕ ЗАДАЧИ КОШИ ДЛЯ ОБЫКНОВЕННЫХ ДИФФЕРЕНЦИАЛЬНЫХ УРАВНЕНИЙ

Представлены описание и анализ метода Рунге — Кутты и линейных многошаговых разностных методов для решения одного ОДУ и систем таких уравнений. Даны основные понятия, позволяющие характеризовать точность и эффективность методов. Введено понятие жесткой системы ОДУ. Представлены методы решения таких систем и связанные с их применением понятия.

7.1. Постановка задачи

Рассмотрим *задачу Коши* для ОДУ n -го порядка в форме, разрешенной относительно старшей производной:

$$u^{(n)} = f(t, u, u', \dots, u^{(n-1)}); \\ t = t_0: u = u_0; u' = u'_0, \dots, u^{(n-1)} = u_0^{(n-1)},$$

где $u = u(t)$ — неизвестная функция; t — независимая переменная. Как правило, независимая переменная имеет смысл времени, и тогда ОДУ описывает эволюцию функции u . Возможны, однако, и другие математические модели, приводящие к задачам Коши, где независимая переменная может иметь другой смысл и другое обозначение.

Хорошо известно, что с помощью замены $u_k = u^{(k-1)}$, $k = \overline{1, n}$, задачу для ОДУ n -го порядка можно свести к системе из n ОДУ первого порядка:

$$u'_k(t) = u_{k+1}(t), k = \overline{1, n-1}; \quad u'_n(t) = f(t, u_1, \dots, u_n), \\ t = t_0: u_k(t_0) = u_0^{(k-1)}, \quad k = \overline{1, n}.$$

Записанная система — частный случай системы ОДУ первого порядка в так называемой нормальной форме:

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

где u и f — вектор-функции из n компонент.

Тогда исходную задачу можно переписать в виде

$$u' = f(t, u);$$

$$u(t_0) = u_0.$$

Параметр t_0 , означающий стартовый момент расчета (зачастую $t_0 = 0$), и вектор u_0 , задающий начальное условие, считаются известными.

Будем изучать численные методы решения такой задачи, полагая, что условия существования и единственности ее решения выполнены. К таким условиям относятся следующие.

1. Пусть функция $f(t, u)$ определена и непрерывна в прямоугольнике

$$D = \{(t, u) : |t - t_0| \leq a; |u_i - u_{0,i}| \leq b, i = \overline{1, n}\}.$$

В этом случае в прямоугольнике D все компоненты $|f_i| \leq M$.

2. Пусть функция $f(t, u)$ липшиц-непрерывна с постоянной L по переменным u_1, u_2, \dots, u_n :

$$\left| f(t, u^{(1)}) - f(t, u^{(2)}) \right| \leq L \sum_{i=1}^n |u_i^{(1)} - u_i^{(2)}|.$$

Тогда существует единственное решение задачи Коши на участке

$$|t - t_0| \leq \tilde{t} = \min \{a, b/M, 1/L\}.$$

Далее большинство методов численного решения будет излагаться на примере задачи Коши для одного уравнения. Как правило, методы для одного уравнения обобщаются на случай систем. Однако будут рассматриваться и методы, специально ориентированные на решение систем.

Отметим, что поставленная задача эквивалентна **задаче Коши в интегральной форме**:

$$u(t) = u_0 + \int_{t_0}^t f(\xi, u(\xi)) d\xi.$$

Такая запись позволяет совместить в одном уравнении и начальное условие, и закон изменения решения. Кроме того, она служит основой для конструирования многих методов численного решения ОДУ, получаемых путем аппроксимации интеграла квадратурными формулами.

Простейшие методы решения задачи Коши (метод Эйлера, симметричная схема, метод Рунге — Кутты второго порядка) будут рассмотрены далее на конечном отрезке $t \in [0, T]$, на котором введена равномерная сетка

$$\omega_\tau = \{t_k = k\tau, k = 0, 1, \dots\}.$$

Величина τ называется шагом сетки, а t_k — ее узлами.

7.2. Простейшие методы численного решения задачи Коши

7.2.1. Методы Эйлера

В интегральной форме задачи Коши (см. 7.1) заменим интеграл квадратурной формулой левых прямоугольников и вычислим приближение к значению решения в точке t_1 :

$$u(t_1) \approx y_1 = y_0 + \tau f(t_0, y_0).$$

Здесь и далее нижний индекс указывает на номер точки, в которой вычисляется функция (или ее производная).

Аналогично, заменяя пределы интегрирования на $[t_1, t_2]$, $[t_2, t_3]$ и т. д., можно записать:

$$y_2 = y_1 + \tau f(t_1, y_1); \dots; y_{n+1} = y_n + \tau f(t_n, y_n).$$

Таким образом получим метод определения значения приближенного решения $y_n = y(t_n)$ на сетке ω_τ . Функция y — *сеточная*.

Записав этот метод в стандартном виде разностного уравнения, получим **явный метод Эйлера**, или просто **метод Эйлера**:

$$\frac{y_{n+1} - y_n}{\tau} = f(t_n, y_n), \quad n = 1, 2, \dots; \quad y_0 = u_0.$$

Заменив интеграл по формуле правых прямоугольников, получим **неявный метод Эйлера**:

$$\frac{y_{n+1} - y_n}{\tau} = f(t_{n+1}, y_{n+1}).$$

В отличие от явного метода, здесь для расчета очередного значения приближенного решения необходимо решить, вообще говоря, нелинейное уравнение.

Обсудим точность полученных формул.

Определение. Численный метод решения задачи Коши называется **сходящимся в точке** $t^* = t_n = n\tau$, если

$$|y_n - u(t^*)| \rightarrow 0 \text{ при } \tau \rightarrow 0,$$

где y_n — приближенное решение в данной точке; $u(t^*) = u_n$ — точное решение ($u_n = u(t_n)$). При этом $\{y_n\}$ — последовательность приближенных решений на последовательности сеток $\{\omega_\tau\}$ таких, что $t^* = n\tau$. Метод называется **сходящимся на отрезке** $[0, T]$, если он сходится в каждой точке $[0, T]$. Метод **имеет p -й порядок точности**, если

$$|y_n - u(t_n)| = O(\tau^p) \text{ при } \tau \rightarrow 0.$$

Определение. Погрешностью численного метода решения задачи Коши называется сеточная функция

$$z_n = y_n - u(t_n).$$

Из выражения для y_{n+1} имеем

$$\frac{z_{n+1} - z_n}{\tau} = f(t_n, z_n + u(t_n)) - \frac{u_{n+1} - u_n}{\tau} = \psi_h^{(1)} + \psi_h^{(2)},$$

где

$$\psi_h^{(1)} = f(t_n, u_n) - \frac{u_{n+1} - u_n}{\tau}; \quad \psi_h^{(2)} = f(t_n, z_n + u_n) - f(t_n, u_n).$$

Определение. Сеточная функция $\psi_h^{(1)}$ называется **невязкой** или **погрешностью аппроксимации** разностного уравнения на решении исходного уравнения, а функция $\psi_h^{(2)}$ — **погрешностью аппроксимации правой части уравнения**.

По формуле конечных приращений Лагранжа

$$\psi_h^{(2)} = \frac{\partial f}{\partial u}(t_n, u_n + \theta z_n) z_n,$$

где $\theta \in (0; 1)$.

Невязка может быть вычислена в результате подстановки точного решения в разностное уравнение. Если $y_n = u_n$, то невязка обращается в нуль.

Говорят, что имеет место **аппроксимация разностного метода на точном решении** исходного дифференциального уравнения, если при $\tau \rightarrow 0$

$$|\psi_h^{(1)}| \rightarrow 0.$$

Если же

$$|\psi_h^{(1)}| = O(\tau^p),$$

то говорят, что имеет место **аппроксимация разностного метода p -го порядка**.

Здесь и далее будем полагать, что все производные нужного порядка существуют. Тогда

$$\psi_h^{(1)} = f(t_n, u_n) - u'(t_n) - \frac{1}{2}u''(t_n + \tilde{\theta}\tau)\tau = O(\tau)$$

в случае ограниченной второй производной u'' , так как на точном решении справедливо равенство $f(t_n, u_n) = u'(t_n)$.

Замечание 7.1. Метод Эйлера встречается в качественной теории обыкновенных дифференциальных уравнений при доказательстве теоремы существования решения задачи Коши с помощью так называемых ломаных Эйлера.

7.2.2. Симметричная схема

В интегральной форме задачи Коши (см. 7.1) заменим интеграл формулой трапеций:

$$y_{n+1} = y_n + \frac{\tau}{2} [f(t_n, y_n) + f(t_{n+1}, y_{n+1})].$$

Получим следующее разностное уравнение:

$$\frac{y_{n+1} - y_n}{\tau} = \frac{f(t_n, y_n) + f(t_{n+1}, y_{n+1})}{2}.$$

Оно, вообще говоря, нелинейное, так как для нахождения решения y_{n+1} в новой временной точке требуется решить нелинейное уравнение. Для этой схемы на трижды непрерывно дифференцируемых решениях погрешность аппроксимации

$$\begin{aligned} \psi_h^{(1)} &= \frac{f(t_n, u_n) + f(t_{n+1}, u_{n+1})}{2} - \frac{u_{n+1} - u_n}{\tau} = \\ &= \frac{u'_n + u'_{n+1}}{2} - \frac{u_{n+1} - u_n}{\tau} = \\ &= \frac{1}{2} \left[u'_{n+1/2} - \frac{\tau}{2} u''_{n+1/2} + \frac{\tau^2}{8} u'''(\eta_1) + \right. \\ &\quad \left. + u'_{n+1/2} + \frac{\tau}{2} u''_{n+1/2} + \frac{\tau^2}{8} u'''(\eta_2) \right] - \\ &- \frac{1}{\tau} \left[u_{n+1/2} + \frac{\tau}{2} u'_{n+1/2} + \frac{\tau^2}{8} u''_{n+1/2} + \frac{\tau^3}{48} u'''(\eta_3) - \right. \\ &\quad \left. - u_{n+1/2} + \frac{\tau}{2} u'_{n+1/2} - \frac{\tau^2}{8} u''_{n+1/2} + \frac{\tau^3}{48} u'''(\eta_4) \right] = O(\tau^2), \end{aligned}$$

где η_1, \dots, η_4 — точки для соответствующих остаточных членов формулы Тейлора в форме Лагранжа. Здесь и далее

обозначения u'_n , $u'_{n+1/2}$, $u'''_{n+1/2}$ и им подобные использованы для значений соответствующих производных в указанных точках сетки. Индекс $n + 1/2$ указывает на середину отрезка между n -й и $(n + 1)$ -й точками: $t_{n+1/2} = (t_n + t_{n+1})/2$.

7.2.3. Метод Рунге — Кутты второго порядка

Метод Эйлера и *симметричная схема* — простейшие примеры так называемых разностных схем. *Методы Рунге — Кутты* в отличие от них требуют вычисления правой части уравнения не только в точках сетки, но и в промежуточных точках.

Для использования формулы центральных прямоугольников при интегрировании правой части уравнения необходимо знать значение приближенного решения в точке $t_{n+1/2}$. Для вычисления этого приближения выполним сначала один шаг по схеме Эйлера с шагом $\tau/2$:

$$y_{n+1/2} = y_n + \frac{\tau}{2} f(t_n, y_n).$$

Далее вычислим y_{n+1} :

$$y_{n+1} = y_n + \tau f(t_{n+1/2}, y_{n+1/2}).$$

Погрешность аппроксимации полученной схемы

$$\begin{aligned} \psi_h^{(1)} &= f\left(t_n + \frac{\tau}{2}, u_n + \frac{\tau}{2} f(t_n, u_n)\right) - \frac{1}{\tau} (u_{n+1} - u_n) = \\ &= f(t_n, u_n) + \frac{\tau}{2} [f'_t(t_n, u_n) + f(t_n, u_n) f'_u(t_n, u_n)] + \\ &\quad + O(\tau^2) - u'_n - \frac{\tau}{2} u''_n = \\ &= u'_n + \frac{\tau}{2} u''_n + O(\tau^2) - u'_n - \frac{\tau}{2} u''_n = O(\tau^2), \end{aligned}$$

так как из уравнения $u' = f(t, u)$ следует равенство $u'' = f'_t + f'_u u'$ и далее $u'' = f'_t + f'_u f$.

Таким образом, данный метод имеет второй порядок аппроксимации и в отличие от симметричной схемы является явным, т. е. для нахождения решения в новой временной точке не требуется решать нелинейное уравнение.

Реализация этого метода в виде двух шагов называется **методом «предиктор — корректор»** (предсказание — исправление). Реализация того же метода в виде

$$k_1 = f(t_n, y_n); \quad k_2 = f\left(t_n + \frac{\tau}{2}, y_n + \frac{\tau}{2}k_1\right); \quad y_{n+1} = y_n + \tau k_2$$

называется **двухстадийным** (или **двухэтапным**) **методом Рунге — Кутты**.

Существуют две большие группы методов численного решения задачи Коши для ОДУ: многошаговые разностные методы и методы Рунге — Кутты. Не все методы естественным образом подлежат такой классификации, но большинство все же принадлежит указанным группам.

7.3. Методы Рунге — Кутты

7.3.1. Явные методы Рунге — Кутты

Рассмотрим задачу Коши для одного дифференциального уравнения:

$$u'_t = f(t, u), \quad t \geq 0;$$

$$u(0) = u_0.$$

Построим метод, позволяющий по значению приближенного решения y_n в точке t_n находить его значение y_{n+1} в точке $t_{n+1} = t_n + \tau$. Учтем также эквивалентную интегральную форму задачи Коши в виде

$$u(t_{n+1}) = u(t_n) + \int_{t_n}^{t_{n+1}} f(t, u) dt.$$

Для получения метода высокого порядка точности необходимо аппроксимировать интеграл в правой части последнего равенства квадратурной формулой соответствующей точности, что влечет необходимость вычислять подынтегральное выражение $f(t, u)$ не только в точках сетки ω_τ , но и в промежуточных точках.

Явный m -стадийный метод Рунге — Кутты заключается в задании коэффициентов a_i , b_{ij} и σ_i , $i = \overline{1, m}$; $j = \overline{1, i-1}$, и последовательном вычислении значений правой части ОДУ:

$$\begin{aligned} k_1 &= f(t_n, y_n); \\ k_2 &= f(t_n + a_2\tau, y_n + b_{21}\tau k_1); \\ k_3 &= f(t_n + a_3\tau, y_n + b_{31}\tau k_1 + b_{32}\tau k_2); \\ \dots &\dots \\ k_m &= f\left(t_n + a_m\tau, y_n + \tau \sum_{j=1}^{m-1} b_{mj} k_j\right) \end{aligned}$$

для дальнейшего нахождения приближенного решения y_{n+1} по формуле

$$y_{n+1} = y_n + \tau \sum_{j=1}^m \sigma_j k_j.$$

Параметры a_i , b_{ij} , σ_j , равно как и m , выбирают таким образом, чтобы имела место аппроксимация метода на точном решении, а также из соображений точности и устойчивости метода.

Видно, что коэффициенты a_i определяют положение дополнительных точек, в которых вычисляется правая часть ОДУ, а b_{ij} и σ_j имеют смысл нормированных на τ весов квадратурных формул, заменяющих интегралы от правой части уравнения по отрезкам $[t_n, t_n + a_i\tau]$ и $[t_n, t_{n+1}]$ соответственно. Из соображений точности формулы для $u(t_{n+1}) \approx y_{n+1}$ при постоянной правой части уравнения необходимо выполнение условия

$$\sum_{j=1}^m \sigma_j = 1.$$

Если потребовать, чтобы и промежуточные значения решения $u(t_n + a_i \tau) \approx y_n + \tau \sum_{j=1}^{i-1} b_{ij} k_j$, $i = \overline{1, m}$, при постоянной правой части ОДУ вычислялись точно, то получим следующие условия, называемые **условиями согласования**:

$$\sum_{j=1}^{i-1} b_{ij} = a_i, \quad i = \overline{1, m}.$$

Рассмотрим различные варианты методов Рунге — Кутты с разным количеством стадий m .

При $m = 1$ получается схема Эйлера в своем обычном виде.

При $m = 2$ имеем семейство методов:

$$\begin{aligned} k_1 &= f(t_n, y_n); \\ k_2 &= f(t_n + a_2 \tau, y_n + b_{21} \tau k_1); \\ y_{n+1} &= y_n + \tau(\sigma_1 k_1 + \sigma_2 k_2). \end{aligned}$$

Для исследования погрешности аппроксимации запишем схему вычисления приближенного решения в виде

$$\frac{y_{n+1} - y_n}{\tau} = \sigma_1 f(t_n, y_n) + \sigma_2 f\left(t_n + a_2 \tau, y_n + b_{21} \tau f(t_n, y_n)\right).$$

Отсюда

$$\begin{aligned} \psi_h^{(1)} &= -\frac{u_{n+1} - u_n}{\tau} + \sigma_1 f(t_n, u_n) + \\ &+ \sigma_2 f\left(t_n + a_2 \tau, u_n + b_{21} \tau f(t_n, u_n)\right) = -u'_n - \frac{1}{2} \tau u''_n + \sigma_1 u'_n + \\ &+ \sigma_2 [u'_n + a_2 \tau f'_t(t_n, u_n) + b_{21} \tau f(t_n, u_n) f'_u(t_n, u_n)] + O(\tau^2), \end{aligned}$$

и так как $u'' = f'_t + f'_u f$, то

$$\begin{aligned} \psi_h^{(1)} &= \tau f'_t(t_n, u_n) \left(\sigma_2 a_2 - \frac{1}{2} \right) + \\ &+ \tau f(t_n, u_n) f'_u(t_n, u_n) \left(\sigma_2 b_{21} - \frac{1}{2} \right) + O(\tau^2). \end{aligned}$$

Использовано также условие $\sigma_1 + \sigma_2 = 1$.

Если все остальные параметры произвольны, то порядок аппроксимации равен единице. Двухстадийные методы второго порядка аппроксимации могут быть получены при $\sigma_2 a_2 = \sigma_2 b_{21} = 1/2$. Обозначим $\sigma = \sigma_2$, $a = a_2 = b_{21}$. Тогда, выбрав $a = 1/(2\sigma)$, методы второго порядка можно записать в виде

$$\frac{y_{n+1} - y_n}{\tau} = (1 - \sigma)f(t_n, y_n) + \sigma f(t_n + a\tau, y_n + a\tau f(t_n, y_n)).$$

Если $\sigma = 1$, то получим метод, рассмотренный в 7.2.3. На практике применяется и метод с $\sigma = 1/2$.

Покажем, что методов Рунге — Кутты третьего порядка аппроксимации при $m = 2$, вообще говоря, не существует. Пусть $f(t, u) = u$, тогда полученный метод (по крайней мере второго порядка аппроксимации) дает

$$\frac{y_{n+1} - y_n}{\tau} = (1 - \sigma)y_n + \sigma(y_n + a\tau y_n) = y_n + \frac{\tau y_n}{2}.$$

При этом

$$\begin{aligned} \psi_h^{(1)} &= \left(1 + \frac{1}{2}\tau\right)u_n - \frac{1}{\tau}(u_{n+1} - u_n) = \\ &= u'_n + \frac{1}{2}\tau u''_n - u'_n - \frac{1}{2}\tau u''_n - \frac{1}{6}\tau^2 \tilde{u}_n''' = -\frac{1}{6}\tau^2 u(t_n + \theta\tau), \end{aligned}$$

где $\tilde{u}_n''' = u'''(t_n + \theta\tau)$, $\theta \in [0; 1]$. Последнее равенство в выражении для $\psi_h^{(1)}$ справедливо, поскольку $u' = u'' = u''' = u$. Таким образом, наивысший порядок аппроксимации равен двум.

На практике в различных пакетах прикладных программ в основном используются методы Рунге — Кутты третьего и четвертого порядков. Обычно эти методы включают соответственно три и четыре стадии, т. е. три или четыре дополнительных вычисления k_i . Увеличение количества стадий повышает вычислительную сложность метода, поскольку расчет значений функции $f(t, u)$ правой части ОДУ зачастую весьма непрост.

Приведем без вывода один из наиболее часто используемых в пакетах прикладных программ **метод Рунге — Кутты четвертого порядка**. Он состоит в последовательном вычислении значений следующих величин:

$$\begin{aligned} k_1 &= f(t_n, y_n); \\ k_2 &= f\left(t_n + \frac{\tau}{2}, y_n + \frac{\tau}{2}k_1\right); \\ k_3 &= f\left(t_n + \frac{\tau}{2}, y_n + \frac{\tau}{2}k_2\right); \\ k_4 &= f(t_n + \tau, y_n + \tau k_3), \end{aligned}$$

при этом приближенное решение y_{n+1} находят из выражения

$$y_{n+1} = y_n + \frac{\tau}{6}(k_1 + 2k_2 + 2k_3 + k_4).$$

Поскольку k_2 и k_3 вычислены в одной и той же временной точке, видно, что последнее выражение есть по существу квадратурная формула Симпсона, а для вычисления значений k_2 , k_3 и k_4 использованы формулы левых, правых и центральных прямоугольников соответственно.

Замечание 7.2. Количество стадий метода Рунге — Кутты не обязательно совпадает с его порядком. Зачастую из соображений устойчивости вычислительного алгоритма удобнее использовать методы, количество стадий в которых превышает порядок метода. Более того, увеличение количества стадий должно опережать увеличение порядка метода, поскольку имеют место так называемые **барьеры Бутчера**. Так, среди явных пятистадийных методов Рунге — Кутты не существует методов пятого порядка аппроксимации (первый барьер Бутчера). С возрастанием количества m стадий увеличивается и разрыв между m и порядком метода. В то же время количество стадий метода Рунге — Кутты напрямую связано с его вычислительной трудоемкостью, поскольку сложность расчета определяется сложностью функции $f(t, u)$ в правой части ОДУ.

Часто для сокращенной записи методов Рунге — Кутты используют так называемые **таблицы Бутчера**:

0					
a_2	b_{21}				
a_3	b_{31}	b_{32}			
\dots	\dots	\dots	\dots		
a_m	b_{m1}	b_{m2}	\dots	$b_{m,m-1}$	
	σ_1	σ_2	\dots	σ_{m-1}	σ_m

В частности, методы второго и четвертого порядков могут быть записаны следующим образом:

а) двухшаговый метод Рунге — Кутты

0				
$1/2$	$1/2$			
	0	1		

б) «классический» метод четвертого порядка

0				
$1/2$	$1/2$			
$1/2$	0	$1/2$		
1	0	0	1	
	$1/6$	$2/6$	$2/6$	$1/6$

в) метод «трех восьмых» четвертого порядка

0				
$1/3$	$1/3$			
$2/3$	$-1/3$	1		
1	1	-1	1	
	$1/8$	$3/8$	$3/8$	$1/8$

Как видно, использование в четырехстадийном методе Рунге — Кутты для вычисления y_{n+1} квадратурной формулы

максимального порядка может приводить и к отрицательным весам в промежуточных квадратурных формулах.

7.3.2. Доказательство сходимости методов Рунге — Кутты

В соответствии с методом Рунге — Кутты

$$\frac{y_{n+1} - y_n}{\tau} = \sum_{j=1}^m \sigma_j k_j,$$

$$k_j = f \left(t_n + a_j \tau, y_n + \sum_{i=1}^{j-1} b_{ji} \tau k_i \right), \quad j = \overline{1, m}; \quad a_1 = 0.$$

Запишем приближенное решение в виде $y_n = u_n + z_n$, где u_n — точное решение; z_n — погрешность. Тогда

$$\frac{z_{n+1} - z_n}{\tau} = \psi_h^{(1)} + \psi_h^{(2)}.$$

Здесь

$$\begin{aligned} \psi_h^{(1)} &= \sum_{i=1}^m \sigma_i k_i(t_n, u_n, \tau) - \frac{u_{n+1} - u_n}{\tau}; \\ \psi_h^{(2)} &= \sum_{i=1}^m \sigma_i [k_i(t_n, y_n, \tau) - k_i(t_n, u_n, \tau)], \end{aligned}$$

где $\psi_h^{(1)}$ — погрешность аппроксимации ОДУ на точном решении (невязка).

Полагаем, что $y_0 = u_0$, т. е. начальные данные задаются точно, $t \in (0, T)$, $t_n = n\tau \leqslant T$ для произвольного n .

Теорема 7.1. Пусть правая часть ОДУ удовлетворяет условию Липшица по второму аргументу с постоянной L , $\psi_h^{(1)}$ — погрешность аппроксимации метода Рунге — Кутты

на точном решении ОДУ (невязка). Тогда для погрешности метода Рунге — Кутты при $n\tau \leq T$ справедлива оценка

$$|z_n| = |y_n - u(t_n)| \leq Te^{\alpha T} \max_{0 \leq j \leq n-1} |\psi_j^{(1)}|.$$

Здесь

$$\alpha = \sigma L m (1 + L b \tau)^{m-1},$$

где $\sigma = \max_{1 \leq i \leq m} |\sigma_i|$; $b = \max_{\substack{1 \leq i \leq m \\ 1 \leq j \leq i-1}} |b_{ij}|$.

◀ Из соотношений для k_i имеем

$$\begin{aligned} |k_i(t_n, y_n, \tau) - k_i(t_n, u_n, \tau)| &\leq \\ &\leq L \left(|y_n - u_n| + \sum_{j=1}^{i-1} \tau |b_{ij}| |k_j(t_n, y_n, \tau) - k_j(t_n, u_n, \tau)| \right), \quad i = \overline{1, m}. \end{aligned}$$

В частности, $|k_1(t_n, y_n, \tau) - k_1(t_n, u_n, \tau)| \leq L|y_n - u_n|$. Введем обозначения:

$$g = L|y_n - u_n| = L|z_n|; \quad r_i = |k_i(t_n, y_n, \tau) - k_i(t_n, u_n, \tau)|.$$

Тогда, используя коэффициент b , имеем

$$r_i \leq g + L\tau b \sum_{j=1}^{i-1} r_j.$$

Получаем

$$r_1 \leq g;$$

$$r_2 \leq g(1 + L\tau b);$$

$$r_3 \leq g + L\tau b g + L\tau b g(1 + L\tau b) = g(1 + L\tau b)^2.$$

Допустим, что для некоторого i справедлива оценка $r_i \leq g\rho^{i-1}$, $\rho = 1 + L\tau b$. Тогда из неравенства $r_i \leq g + L\tau b \sum_{j=1}^{i-1} r_j$ получаем

$$r_{i+1} \leq g + (\rho - 1) \sum_{j=1}^i g\rho^{j-1} = g + (\rho - 1)g \frac{1 - \rho^i}{1 - \rho} = g(1 - 1 + \rho^i) = g\rho^i.$$

Отсюда видно, что для любого i справедлива оценка $r_i \leq g\rho^{i-1}$, $\rho = 1 + L\tau b$. Следовательно,

$$\begin{aligned} |\psi_h^{(2)}| &\leq \sum_{j=1}^m |\sigma_j| |r_j| \leq \sigma g \sum_{j=1}^m \rho^{j-1} \leq \sigma g m \rho^{m-1} \leq \\ &\leq \sigma L |z_n| m (1 + \tau L b)^{m-1} = \alpha |z_n| \end{aligned}$$

(здесь использованы введенные в формулировке теоремы параметры α и σ). Тогда

$$\begin{aligned} |z_{n+1}| &\leq |z_n|(1 + \alpha\tau) + \tau |\psi_{h,n}^{(1)}| \leq \\ &\leq (1 + \alpha\tau)^{n+1} |z_0| + \sum_{j=0}^n \tau (1 + \alpha\tau)^{n-j} |\psi_{h,j}^{(1)}|. \end{aligned}$$

Учитывая, что $z_0 = 0$, имеем

$$\begin{aligned} |z_{n+1}| &\leq (n+1)\tau (1 + \alpha\tau)^n \max_j |\psi_{h,j}^{(1)}| \leq \\ &\leq t_{n+1} e^{\alpha t_n} \max_{1 \leq j \leq n} |\psi_h^{(1)}| \leq T e^{\alpha T} \max_{1 \leq j \leq n} |\psi_{h,j}^{(1)}|. \quad \blacktriangleright \end{aligned}$$

Следствие 7.1. При выполнении условий теоремы 7.1 порядок точности метода Рунге — Кутты совпадает с порядком аппроксимации.

◀ Это следует из доказанной оценки погрешности и равномерной по τ ограниченности α :

$$\alpha = \sigma L m (1 + L b \tau)^{m-1} \leq \sigma L m e^{(m-1)Lb\tau} \leq \sigma L m e^{(m-1)LbT}. \quad \blacktriangleright$$

Замечание 7.3. Если принять $T = \tau$ и $n = 1$, то в условиях теоремы 7.1 оценка погрешности, допущенной на одном шаге метода Рунге — Кутты, имеющего порядок аппроксимации p , при $y_0 = u(t_0)$ составляет

$$y_1 - u(t_0 + \tau) = O(\tau^{p+1}).$$

Этот факт можно соотнести с погрешностями частичных и полных квадратурных формул, рассмотренных в предыдущей главе.

7.3.3. Управление длиной шага

Оценка погрешности численного решения, представленная в доказательстве сходимости методов Рунге — Кутты (см. 7.3.2), весьма громоздка и вряд ли применима для практической оценки точности полученных численных результатов. В то же время для аккуратного выполнения расчетов оценка погрешностей необходима, чтобы обеспечить длину шага τ , с одной стороны, достаточно малую для достижения требуемой точности вычисляемых результатов, а с другой стороны, достаточно большую во избежание бесполезной вычислительной работы. При этом с практической точки зрения важно иметь инструмент управления длиной шага τ в процессе расчета, чтобы корректировать точность получаемого приближенного решения без его полного пересчета. Для этого удобно воспользоваться соображениями, приведенными в замечании 7.3.

Действительно, если необходимо находить приближенное решение задачи Коши, совершая на каждом шаге ошибку не более некоторого наперед заданного ε , можно действовать следующим образом. Пусть используемый метод имеет порядок аппроксимации p . Если при некотором шаге τ_{old} получено приближенное решение, для которого оценка погрешности составляет $\tilde{\varepsilon} \sim \tau_{old}^{p+1}$, то можно выбрать новый шаг

$$\tau_{new} \sim \tau_{old} \left(\frac{\varepsilon}{\tilde{\varepsilon}} \right)^{\frac{1}{p+1}}.$$

Задача заключается в поиске оценки погрешности решения $\tilde{\varepsilon}$.

Для оценки погрешности численного решения можно воспользоваться идеей **экстраполяции по Ричардсону**, или **правилом Рунге** (а также его обобщением — **процессом Эйткена**).

Пусть $y_0 = u(t_0)$, вычисления проводятся с шагом τ и в точке $t^* = q\tau$, $q \in \mathbb{N}$, получено приближенное значение решения $y_q^{(\tau)}$.

Выполняя вычисления с шагом $q\tau$, можно получить приближенное значение решения $y_1^{(q\tau)}$ в той же точке. Для метода порядка p в силу замечания 7.3 можем записать систему уравнений

$$y_q^{(\tau)} = u(t^*) + qC\tau^{p+1} + O(\tau^{p+2});$$

$$y_1^{(q\tau)} = u(t^*) + C(q\tau)^{p+1} + O(\tau^{p+2}).$$

Решая ее с точностью $O(\tau^{p+2})$, запишем

$$y_q^{(\tau)} = u(t^*) + \frac{y_1^{(q\tau)} - y_q^{(\tau)}}{q^p - 1} + O(\tau^{p+2})$$

и получим оценку погрешности решения $y_q^{(\tau)}$:

$$\tilde{\varepsilon} \approx \left| \frac{y_1^{(q\tau)} - y_q^{(\tau)}}{q^p - 1} \right|.$$

Отметим, что полученные соотношения имеют асимптотический смысл. Тем не менее правило Рунге широко применяется для расчетов, в частности, для автоматического выбора шага в программных комплексах.

Как правило, наиболее простой вариант выбора q для оценки погрешности — это $q = 2$. В этом случае если на очередном шаге оказалось, что погрешность решения превышает заданный порог $\tilde{\varepsilon} > \varepsilon$, то последние два шага должны быть отброшены, а длина шага τ скорректирована. Увеличение q ведет к увеличению количества отбрасываемых шагов.

Другой способ построения оценки погрешности численного решения заключается в том, чтобы использовать такие формулы Рунге — Кутты, которые позволяли бы вычислить вместе со значением приближенного решения y_{n+1} более точное (более высокого порядка) приближение \hat{y}_{n+1} , затратив при этом минимальные вычислительные ресурсы. Этого можно добиться, если использовать пару методов Рунге — Кутты, внутренние

стадии которых совпадают (по крайней мере, частично). Иными словами, требуется построить пару методов с таблицей Бутчера

0					
a_2	b_{21}				
a_3	b_{31}	b_{32}			
\dots	\dots	\dots	\dots		
a_m	b_{m1}	b_{m2}	\dots	$b_{m,m-1}$	
	σ_1	σ_2	\dots	σ_{m-1}	σ_m
	$\hat{\sigma}_1$	$\hat{\sigma}_2$	\dots	$\hat{\sigma}_{m-1}$	$\hat{\sigma}_m$

так, чтобы метод

$$y_{n+1} = y_n + \tau(\sigma_1 k_1 + \dots + \sigma_m k_m)$$

имел порядок p , а метод

$$\hat{y}_{n+1} = y_n + \tau(\hat{\sigma}_1 k_1 + \dots + \hat{\sigma}_m k_m),$$

например, порядок $q = p + 1$. Тогда можно получить оценку погрешности решения y_{n+1}

$$\tilde{\varepsilon} = |y_{n+1} - \hat{y}_{n+1}|.$$

Такие методы получили название **вложенных методов Рунге — Кутты** порядка $p(q)$. Например, таблица Бутчера

0					
1		1			
1/2	1/4	1/4			
y_{n+1}	1/2	1/2	0		
\hat{y}_{n+1}	1/6	1/6	4/6		

задает метод Фельберга порядка 2(3). Отметим, что значение приближенного решения y_{n+1} получается путем использования квадратурной формулы трапеций, для чего вычисляются значения правой части ОДУ k_1 и k_2 в левой и правой точках отрезка $[t_n, t_{n+1}]$. При этом значение \hat{y}_{n+1} получено по более точной квадратурной формуле Симпсона. Для ее применения необходимо

дополнительно вычислить значение функции правой части ОДУ в центре отрезка $[t_n, t_{n+1}]$, что и реализует вычисление k_3 . При этом точность вычисления k_3 также повышена, поскольку для него используется среднее значение функции правой части уравнения в точках t_n и t_{n+1} .

Отметим, что использование для дальнейших расчетов в качестве значения приближенного решения величины \hat{y}_{n+1} нежелательно, несмотря на ее потенциально более высокую точность. Дело в том, что в таком случае $\tilde{\varepsilon}$ уже не является оценкой погрешности приближенного решения, в связи с чем эта оценка может быть существенно занижена.

Замечание 7.4. Применение правила Рунге для оценки погрешности увеличивает вычислительные затраты при выполнении двух шагов в 1,5 раза, в то время как при использовании вложенных методов Рунге — Кутты часто требуется вычисление лишь одного дополнительного значения k_i на каждом временному шаге. Однако правило Рунге универсально — его можно применять вместе с любым базовым методом Рунге — Кутты.

С использованием любого из описанных способов оценки погрешности можно построить алгоритм автоматического выбора шага τ_n для метода порядка p . Пусть $\tilde{\varepsilon}$ — оценка погрешности численного решения на шаге n . Тогда длина следующего шага может быть вычислена по формуле

$$\tau_{n+1} = \tau_n \min \left(F_{\max}, \max \left(F_{\min}, F \left(\frac{\varepsilon}{\tilde{\varepsilon}} \right)^{\frac{1}{p+1}} \right) \right),$$

где ε — желаемое значение погрешности; F_{\max} — максимальный коэффициент увеличения длины шага (обычно $F_{\max} \in [1,5; 5]$); F_{\min} — минимальный коэффициент изменения шага; F — коэффициент запаса, учитывающий неточность оценки погрешности $\tilde{\varepsilon}$ (обычно $F \in [0,8; 0,9]$). Коэффициенты F_{\max} и F_{\min} позволяют

стабилизировать алгоритм выбора шага. Все три коэффициента F , F_{\max} и F_{\min} задаются из соображений точности и устойчивости алгоритма и являются настроочными. Отметим, что если $\tilde{\varepsilon} > \varepsilon$, то для обеспечения точности приближенного решения необходимо отбросить результаты расчета последнего шага (в случае применения вложенных методов Рунге — Кутты) или двух последних шагов (в случае использования правила Рунге с $q = 2$).

7.4. Многошаговые разностные методы

7.4.1. Определение линейных многошаговых методов

Рассмотрим снова задачу Коши для одного ОДУ:

$$u' = f(t, u), \quad t \geq 0,$$

$$u(0) = u_0$$

и сетку ω_τ с постоянным шагом $\tau > 0$.

Линейный m -шаговый разностный метод решения ОДУ задается системой разностных уравнений:

$$\frac{a_0 y_n + a_1 y_{n-1} + \dots + a_m y_{n-m}}{\tau} = b_0 f_n + b_1 f_{n-1} + \dots + b_m f_{n-m},$$

$$n = m, m+1, \dots.$$

Как правило, удобно принять $f_{n-k} = f(t_{n-k}, y_{n-k})$.

Линейный m -шаговый метод позволяет получить значения приближенного решения ОДУ, начиная с узла сетки с номером $n = m$. Следовательно, для обеспечения старта расчета необходимо задать y_0, y_1, \dots, y_{m-1} . Обычно значения этих величин при постановке задачи неизвестны, но их можно вычислить с помощью каких-либо иных методов, например методов Рунге — Кутты.

В рассматриваемом методе $a_0 \neq 0$, числовые коэффициенты a_i , b_i , $i = \overline{0, m}$, определены с точностью до постоянного сомножителя.

Если сумма коэффициентов b_i равна единице, т. е. при выполнении условия

$$\sum_{i=0}^m b_i = 1,$$

правая часть разностного уравнения может аппроксимировать функцию в правой части исходного ОДУ. Это условие используют в качестве нормировочного для однозначного задания коэффициентов $a_i, b_i, i = \overline{0, m}$.

Если $b_0 = 0$, то линейный m -шаговый разностный **метод решения задачи Коши** называется **явным**, в противном случае — **неявным**. Если $b_k = 0, k = \overline{1, m}, b_0 = 1$, то такой метод часто называют **полностью неявным**.

В случае неявного метода для нахождения y_n необходимо решать, вообще говоря, нелинейное уравнение.

Рассмотрим **два способа построения линейных многошаговых методов**.

Первый способ опирается на следующую идею. Для точного решения задачи Коши справедливо равенство

$$u(t_{n+1}) = u(t_n) + \int_{t_n}^{t_{n+1}} f(t, u(t)) dt.$$

Пусть известны значения f_{n-k} правой части уравнения при $k = \overline{0, m}$. Построим с их помощью интерполяционный полином

$$Q_m^{(n)}(t) = \sum_{k=0}^m f_{n-k} \varphi_k^n(t),$$

где $\varphi_k^n(t)$ — базисные функции лагранжевой интерполяции:

$$\varphi_k^n(t) = \prod_{\substack{i=0, \\ i \neq k}}^m \frac{t - t_{n-i}}{t_{n-k} - t_{n-i}}.$$

Подставим полином $Q_m^{(n)}(t)$ под знак интеграла, заменив значения точного решения $u(t_n)$ приближенными y_n . Тогда

$$y_{n+1} = y_n + \int_{t_n}^{t_{n+1}} Q_m^{(n)}(t) dt = y_n + \sum_{k=0}^m f_{n-k} \int_{t_n}^{t_{n+1}} \varphi_k^n(t) dt.$$

Получим явный линейный многошаговый метод вида

$$\frac{y_{n+1} - y_n}{\tau} = \sum_{k=0}^m b_k f_{n-k},$$

$$\text{где } b_k = \frac{1}{\tau} \int_{t_n}^{t_{n+1}} \varphi_k^n(t) dt.$$

Второй способ опирается на тот факт, что ОДУ выполнено во всех точках сетки ω_τ , в частности в точке t_n , т. е.

$$u'(t_n) = f(t_n, u(t_n)).$$

Используя $m+1$ значений приближенного решения y_{n-m}, \dots, y_n , построим интерполяционный полином

$$P_m^{(n)}(t) = \sum_{k=0}^m y_{n-k} \varphi_k^n(t)$$

и потребуем выполнения равенства

$$\left. \frac{d}{dt} P_m^{(n)}(t) \right|_{t=t_n} = f(t_n, y_n),$$

или

$$\sum_{k=0}^m y_{n-k} (\varphi_k^n(t))'_{t=t_n} = f(t_n, y_n).$$

Получим неявный линейный многошаговый метод вида

$$\frac{1}{\tau} \sum_{k=0}^m a_k y_{n-k} = f_n,$$

$$\text{где } a_k = \tau (\varphi_k^n(t))'_{t=t_n}; f_n = f(t_n, y_n).$$

Оба варианта построения многошаговых разностных методов можно легко модифицировать для случая переменного шага τ .

Определение. Линейный m -шаговый метод решения задачи Коши с $a_0 = -a_1 = 1$, $a_k = 0$, $k = \overline{2, m}$, называется **явным методом Адамса**:

$$\frac{y_{n+1} - y_n}{\tau} = \sum_{k=0}^m b_k f_{n-k}.$$

Определение. Неявный линейный m -шаговый метод решения задачи Коши с $b_0 = 1$, $b_k = 0$, $k = \overline{1, m}$, называется **методом Гира**:

$$\frac{1}{\tau} \sum_{k=0}^m a_k y_{n-k} = f_n.$$

Методы Гира часто называют также **формулами дифференцирования назад** (ФДН). Порядок метода определяется тем, с каким порядком сеточный оператор

$$D_h[y] = \frac{1}{\tau} \sum_{k=0}^m a_k y_{n-k}$$

аппроксимирует оператор первой производной

$$D[u] = u'.$$

Методы Гира неявные, поэтому на каждом шаге приводят к решению, вообще говоря, нелинейного уравнения.

7.4.2. Погрешность аппроксимации многошаговых методов

Для погрешности аппроксимации линейного многошагового метода справедливо выражение

$$\psi_h^{(1)} = \sum_{k=0}^m b_k f(t_{n-k}, u_{n-k}) - \frac{1}{\tau} \sum_{k=0}^m a_k u_{n-k}.$$

Предположив, что решение имеет производные нужного порядка, получим

$$u_{n-k} = u(t_n - k\tau) = \sum_{l=0}^p \frac{(-k\tau)^l u^{(l)}(t_n)}{l!} + O(\tau^{p+1});$$

$$f(t_{n-k}, u_{n-k}) = u'_{n-k} = u'(t_n - k\tau) = \sum_{l=0}^{p-1} \frac{(-k\tau)^l u^{(l+1)}(t_n)}{l!} + O(\tau^p).$$

В результате имеем

$$\begin{aligned} \psi_h^{(1)} &= - \sum_{k=0}^m \frac{1}{\tau} a_k \sum_{l=0}^p \frac{(-k\tau)^l u^{(l)}(t_n)}{l!} + \\ &+ \sum_{k=0}^m b_k \sum_{l=0}^{p-1} \frac{(-k\tau)^l u^{(l+1)}(t_n)}{l!} + O(\tau^p) = - \left(\sum_{k=0}^m \frac{a_k}{\tau} \right) u(t_n) + \\ &+ \sum_{l=1}^p \sum_{k=0}^m \left(b_k \frac{(-k\tau)^{l-1}}{(l-1)!} - \frac{1}{\tau} a_k \frac{(-k\tau)^l}{l!} \right) u^{(l)}(t_n) + O(\tau^p). \end{aligned}$$

Следовательно, для обеспечения погрешности аппроксимации p -го порядка должны выполняться следующие условия:

$$\sum_{k=0}^m \frac{1}{\tau} a_k = 0; \quad \sum_{k=0}^m k^{l-1} \left(b_k + a_k \frac{k}{l} \right) = 0, \quad l = \overline{1, p}.$$

Условие нормировки

$$\sum_{k=0}^m b_k = 1$$

и полученные $p+1$ уравнений дают систему из $p+2$ уравнений для определения $2(m+1)$ неизвестных $a_i, b_i, i = \overline{0, m}$. Эта система уравнений не является переопределенной только при $2(m+1) \geq p+2$, т. е. $p \leq 2m$. Таким образом, порядок аппроксимации m -шаговых методов не может быть выше $2m$. Очевидно, что для явных методов он не может быть выше $2m-1$.

Для методов Адамса получаем уравнения

$$l \sum_{k=0}^m k^{l-1} b_k = 1, \quad l = \overline{1, p}.$$

Таким образом, имеется p уравнений (вместе с условием нормировки) для определения $m + 1$ параметров. Количество уравнений равно p , так как при $l = 1$ уравнение $l \sum_{k=0}^m k^{l-1} b_k = 1$ совпадает с условием нормировки. Следовательно, в общем случае $p \leq m + 1$. Явные методы Адамса ($b_0 = 0$) имеют порядок аппроксимации $p \leq m$.

Для методов Гира условия, обеспечивающие погрешность аппроксимации p -го порядка, имеют вид

$$\begin{aligned} \sum_{k=0}^m a_k &= 0; \\ \sum_{k=0}^m k^{l-1} \left(b_k + a_k \frac{k}{l} \right) &= 0, \quad l = \overline{1, p}; \\ b_0 &= 1, \quad b_i = 0, \quad i = \overline{1, m}. \end{aligned}$$

Отсюда следует, что

$$\sum_{k=0}^m a_k = 0; \quad \sum_{k=0}^m a_k k = -1; \quad \sum_{k=0}^m k^l a_k = 0, \quad l = \overline{2, p},$$

т. е. имеется $p + 1$ уравнений для $m + 1$ неизвестных. Таким образом, $p \leq m$, как и в случае явных методов Адамса.

7.4.3. Примеры методов Адамса и Гира

Методы Адамса. Рассмотрим примеры построения явных методов Адамса с $b_0 = 0$. Условия p -го порядка аппроксимации имеют вид

$$\sum_{k=0}^m k^{l-1} b_k = l^{-1}, \quad l = \overline{1, p}, \quad p = m.$$

При $m = 1$ из условий порядка следует, что $b_0 = 0$, $b_1 = 1$, т. е. получим метод Эйлера:

$$\frac{y_n - y_{n-1}}{\tau} = f_{n-1}.$$

При $m = 2$

$$b_0 = 0; \quad b_1 + b_2 = 1; \quad b_1 + 2b_2 = 1/2,$$

откуда $b_0 = 0$; $b_1 = 3/2$; $b_2 = -1/2$. Метод Адамса примет вид

$$\frac{y_n - y_{n-1}}{\tau} = \frac{3}{2}f_{n-1} - \frac{1}{2}f_{n-2}.$$

При $m = 3$ получим систему

$$\begin{aligned} b_0 &= 0; \\ b_1 + b_2 &= 1; \\ b_1 + 2b_2 + 3b_3 &= 1/2; \\ b_1 + 4b_2 + 9b_3 &= 1/3, \end{aligned}$$

откуда $b_0 = 0$; $b_1 = 23/12$; $b_2 = -16/12$; $b_3 = 5/12$. Метод Адамса примет вид

$$\frac{y_n - y_{n-1}}{\tau} = \frac{23}{12}f_{n-1} - \frac{16}{12}f_{n-2} + \frac{5}{12}f_{n-3}.$$

Каждый раз при увеличении m получаем систему $m + 1$ линейных уравнений.

Построим теперь метод Адамса с $m = 2$ по-другому. Пусть известны значения f_{n-1} , f_{n-2} и y_{n-1} . Тогда примем, что

$$y_n = y_{n-1} + \int_{t_{n-1}}^{t_n} \left(f_{n-1} \frac{t - t_{n-2}}{t_{n-1} - t_{n-2}} + f_{n-2} \frac{t - t_{n-1}}{t_{n-2} - t_{n-1}} \right) dt,$$

заменив правую часть в ОДУ линейным интерполянтом. Интеграл в правой части этого равенства легко вычислить, например, по формуле центральных прямоугольников:

$$\begin{aligned} \int_{t_{n-1}}^{t_n} \left(f_{n-1} \frac{t - t_{n-2}}{\tau} - f_{n-2} \frac{t - t_{n-1}}{\tau} \right) dt &= \\ = \frac{f_{n-1}}{\tau} \tau \left(t_n - \frac{\tau}{2} - t_n + 2\tau \right) - \frac{f_{n-2}}{\tau} \tau \left(t_n - \frac{\tau}{2} - t_n + \tau \right) &= \\ = \frac{3}{2} f_{n-1} - \frac{1}{2} f_{n-2}. & \end{aligned}$$

Снова получим метод, задаваемый формулой

$$\frac{y_n - y_{n-1}}{\tau} = \frac{3}{2} f_{n-1} - \frac{1}{2} f_{n-2}.$$

Если рассматривать и неявные методы Адамса, то $p = m + 1$. Тогда

$$\sum_{k=0}^m k^{l-1} b_k = l^{-1}, \quad l = \overline{1, p}, \quad p = m + 1.$$

При $m = 1$ получим систему

$$b_0 + b_1 = 1;$$

$$b_1 = 1/2,$$

откуда $b_0 = 1/2$, и таким образом приходим к симметричной схеме:

$$\frac{y_{n+1} - y_n}{\tau} = \frac{f_n + f_{n+1}}{2}.$$

Процедура может быть продолжена.

Методы Гира. Условия p -го порядка аппроксимации имеют вид

$$\sum_{k=0}^m a_k = 0; \quad \sum_{k=0}^m a_k k = -1; \quad \sum_{k=0}^m k^l a_k = 0, \quad l = \overline{2, p}.$$

При $m = 1$ из условий порядка следует, что $a_0 + a_1 = 0$ и $a_1 = -1$, откуда получим неявный метод Эйлера:

$$\frac{y_n - y_{n-1}}{\tau} = f_n.$$

При $m = 2$

$$a_0 + a_1 + a_2 = 0;$$

$$a_1 + 2a_2 = -1;$$

$$a_1 + 4a_2 = 0,$$

откуда $a_0 = \frac{3}{2}$; $a_1 = -2$; $a_2 = \frac{1}{2}$, в результате получим метод второго порядка точности

$$\frac{3}{2}y_n - 2y_{n-1} + \frac{1}{2}y_{n-2} = \tau f_n.$$

При $m = 3$ можно построить метод третьего порядка точности. Получим его двумя способами.

1. Условия порядка при $m = 3$ и $p = 3$ дают систему уравнений

$$a_0 + a_1 + a_2 + a_3 = 0;$$

$$a_1 + 2a_2 + 3a_3 = -1;$$

$$a_1 + 4a_2 + 9a_3 = 0;$$

$$a_1 + 8a_2 + 27a_3 = 0.$$

Вычитая второе уравнение системы из третьего и четвертого, получим

$$2a_2 + 6a_3 = 1;$$

$$6a_2 + 24a_3 = 1.$$

Отсюда $a_3 = -\frac{1}{3}$; $a_2 = \frac{3}{2}$; $a_1 = -3$; $a_0 = \frac{11}{6}$. В результате имеем метод Гира третьего порядка в виде

$$\frac{1}{\tau} \left(\frac{11}{6}y_n - 3y_{n-1} + \frac{3}{2}y_{n-2} - \frac{1}{3}y_{n-3} \right) = f_n.$$

2. Пусть известны y_{n-3} , y_{n-2} , y_{n-1} , y_n . Построим интерполяционный полином:

$$\begin{aligned} L_3(t) = & y_{n-3} \frac{(t - t_n + 2\tau)(t - t_n + \tau)(t - t_n)}{(-\tau)(-2\tau)(-3\tau)} + \\ & + y_{n-2} \frac{(t - t_n + 3\tau)(t - t_n + \tau)(t - t_n)}{\tau(-\tau)(-2\tau)} + \\ & + y_{n-1} \frac{(t - t_n + 3\tau)(t - t_n + 2\tau)(t - t_n)}{2\tau \cdot \tau \cdot (-\tau)} + \\ & + y_n \frac{(t - t_n + 3\tau)(t - t_n + 2\tau)(t - t_n + \tau)}{3\tau \cdot 2\tau \cdot \tau}. \end{aligned}$$

Вычислим его производную:

$$L'_3(t_n) = \frac{11}{6\tau}y_n - \frac{3}{\tau}y_{n-1} + \frac{3}{2\tau}y_{n-2} - \frac{1}{3\tau}y_{n-3}.$$

Как и при первом способе, получим метод Гира третьего порядка:

$$\frac{1}{\tau} \left(\frac{11}{6}y_n - 3y_{n-1} + \frac{3}{2}y_{n-2} - \frac{1}{3}y_{n-3} \right) = f_n.$$

Методы Гира с большим количеством шагов редко используют на практике, но их можно получить аналогичным образом.

7.4.4. Устойчивость и сходимость разностных методов

Методы высокого порядка аппроксимации практически не используют, так как они неустойчивы. Приведем без подробностей и доказательств основные понятия, относящиеся к устойчивости и сходимости. Подробно понятие устойчивости численных методов решения дифференциальных уравнений рассматривается в теории разностных схем. Под устойчивостью будем понимать лишь ограниченность или невозрастание численного решения задачи Коши, если такое свойство присуще точному решению.

Логично требовать устойчивости решения, полученного приближенным методом, хотя бы для уравнения $u' = 0$. Покажем,

что, хотя точное решение этого уравнения — постоянная функция, для приближенного решения это не обязательно так.

Выбрав $f(t, u) = 0$, рассмотрим наряду с исходным m -шаговым линейным разностным методом однородное разностное уравнение с постоянными коэффициентами

$$\sum_{k=0}^m a_k y_{n-k} = 0, \quad n = m, m+1, \dots,$$

и будем искать его решение в виде $y_k = q^k$. Тогда для любой точки n в предположении, что $q \neq 0$, получим

$$\sum_{k=0}^m a_k q^{m-k} = 0.$$

Это уравнение называется *характеристическим уравнением линейного m -шагового разностного метода*.

Очевидно, что корни характеристического уравнения q не обязательно равны единице — это единственный вариант, при котором решение разностного уравнения — константа.

Определение. *Линейный m -шаговый разностный метод* решения задачи Коши называется *нуль-устойчивым*, если он удовлетворяет *условию корней*, а именно если все корни q_1, q_2, \dots, q_m характеристического уравнения лежат внутри или на границе единичного круга комплексной плоскости $|q| \leq 1$, причем на границе нет кратных корней.

Отметим, что условие нуль-устойчивости метода обеспечивает лишь невозрастание модуля приближенного решения уравнения $u' = 0$.

Замечание 7.5. Как уже отмечалось в 2.6.1, теория линейных разностных уравнений с постоянными коэффициентами весьма близка к теории линейных ОДУ. В частности, общее решение неоднородного уравнения в обоих случаях можно представить как сумму общего решения однородного уравнения и частного

решения неоднородного уравнения. Общее решение однородного уравнения может быть найдено как линейная комбинация элементарных решений в виде экспоненты в случае ОДУ или степенной зависимости для разностного уравнения. Далее необходимо решить полученное характеристическое уравнение. Каждому простому корню соответствует свое линейно независимое решение. Если корень кратный, то для получения нового элемента фундаментальной системы решений элементарное решение необходимо домножить на степень t в дифференциальном случае или k в разностном случае.

Отсюда ясны причины появления понятия нуль-устойчивости: для устойчивости общего решения необходима устойчивость решения однородного уравнения. Для этого ни одна из функций фундаментальной системы решений, входящих в общее решение, не должна возрастать с увеличением номера k .

Приведем без доказательства две теоремы об устойчивости линейных многошаговых методов.

Теорема 7.2. Пусть разностный m -шаговый метод удовлетворяет условию корней и имеет порядок аппроксимации p . Тогда $p \leq m + 1$ при нечетном m и $p \leq m + 2$ при четном m . Для явных устойчивых m -шаговых методов порядок аппроксимации не превосходит m .

Теорема 7.3. Пусть разностный m -шаговый метод удовлетворяет условию корней и $|f'_y| \leq L$. Тогда для любого $m\tau \leq t_n = n\tau \leq T$ при достаточно малом τ выполнена оценка

$$|y_n - u(t_n)| \leq M \left(\max_{0 \leq j \leq m-1} |y_j - u(t_j)| + \max_{m \leq j \leq n} |\psi_{h,j}^{(1)}| \right),$$

где M — постоянная, не зависящая от m ; $|y_j - u(t_j)|$, $j = \overline{0, m-1}$, — погрешность в задании начальных условий; $\psi_{h,j}^{(1)}$, $j = \overline{m, n}$, — невязка (погрешность аппроксимации).

Из оценки теоремы 7.3 следует сходимость разностного метода, если начальные погрешности сходятся к нулю при $\tau \rightarrow 0$ и имеет место аппроксимация.

Пример 7.1. Методы Адамса всегда удовлетворяют условию корней, так как $a_0 = 1$, $a_1 = -1$, характеристическое уравнение имеет вид $a_0 q + a_1 = 0$, и, следовательно, $q = 1$. •

7.5. Методы решения жестких систем

7.5.1. Условно устойчивые и безусловно устойчивые разностные методы

Условие корней обеспечивает устойчивость только постоянного решения, но не учитывает структуру нетривиальной правой части ОДУ и, следовательно, характерные особенности решения. Рассмотрим, например, следующую задачу Коши:

$$u_t = -\alpha^2 u, \quad t \geq 0;$$

$$u(0) = u_0.$$

Ее решение $u(t) = u_0 e^{-\alpha^2 t}$, откуда $|u(t_{n+1})| < |u(t_n)|$, т. е. решение монотонно (с сохранением знака) убывает.

Для численного решения этой задачи применим метод Эйлера:

$$\frac{y_{n+1} - y_n}{\tau} = -\alpha^2 y_n, \text{ или } y_{n+1} = y_n (1 - \alpha^2 \tau),$$

откуда $|y_{n+1}| \leq |y_n|$ при $0 \leq \tau \leq 2/\alpha^2$. Таким образом, явный метод Эйлера устойчив в смысле удовлетворения оценки $|y_{n+1}| \leq |y_n|$ лишь при выполнении условия $0 < \tau \leq 2/\alpha^2$.

Определение. *Разностный метод* называется *условно устойчивым*, если он устойчив при некоторых ограничениях на шаг τ , и *безусловно устойчивым*, если он устойчив при произвольных τ .

Рассмотрим пример безусловно устойчивого метода — неявный метод Эйлера:

$$\frac{y_{n+1} - y_n}{\tau} = -\alpha^2 y_{n+1},$$

откуда

$$|y_{n+1}| = |y_n(1 + \alpha^2 \tau)^{-1}| \leq |y_n|.$$

Чаще всего явные схемы условно устойчивы, а среди неявных схем есть безусловно устойчивые, т. е. такие, которые не накладывают ограничений на длину шага. Однако явные методы намного проще в реализации, ведь для неявного метода, вообще говоря, придется решать нелинейное уравнение относительно y_{n+1} , если правая часть f уравнения нелинейно зависит от решения.

7.5.2. Понятие жесткой системы ОДУ

Многие рассмотренные ранее методы решения одного ОДУ можно без проблем перенести на случай систем ОДУ. Однако при этом могут появиться и сложности, связанные с тем, что уравнения системы имеют разные свойства.

Пример 7.2. Рассмотрим задачу Коши для системы ОДУ вида

$$u'_1(t) = -u_1(t); \quad u'_2(t) = -\varepsilon^2 u_2(t), \quad t > 0,$$

$$u_1(0) = u_{1,0}; \quad u_2(0) = u_{2,0}.$$

Ее решение $u_1 = u_{1,0} e^{-t}$, $u_2 = u_{2,0} e^{-\varepsilon^2 t}$.

Пусть задача решается на отрезке $[0, T]$, так что $\varepsilon^2 T \gg 1$, например $\varepsilon^2 T = 5$, т. е. $T = 5 \varepsilon^{-2}$. Если для решения системы выбран явный метод Эйлера, то должно быть выполнено условие

$$\tau \leq \min \left\{ 2, \frac{2}{\varepsilon^2} \right\}.$$

Если $\varepsilon^2 \ll 1$, то эти два ограничивающих τ значения различаются в ε^{-2} раз, причем $\varepsilon^{-2} \gg 1$. В результате при расчете на равномерной сетке потребуется $T/\tau = 5/2\varepsilon^{-2} = 2,5\varepsilon^{-2}$ шагов, что

определяется условием устойчивости наиболее быстроизменяющейся компоненты решения. Такое количество шагов становится бессмысленным с некоторого момента времени, поскольку, например, при $t = 5$ значение функции u_1 пренебрежимо мало. •

Рассмотрим более общую ситуацию. Пусть необходимо решить систему $u' = Au$ с постоянной матрицей, которую можно привести к диагональному виду преобразованием $Q^{-1}AQ$. Тогда замена $u = Qv$ приводит исходное уравнение к системе $v' = Q^{-1}AQv$ с диагональной матрицей, которая имеет те же собственные значения, что и матрица A .

Определение. *Система ОДУ* $u' = Au$ с постоянной матрицей $A = A_{m \times m}$ называется **жесткой**, если:

- 1) все собственные значения матрицы A имеют отрицательную действительную часть, т. е. $\operatorname{Re} \lambda_i < 0$, $\lambda_i \in \sigma(A)$, $i = \overline{1, m}$;
- 2) число S , называемое **числом жесткости**, велико:

$$S = \frac{\max_{1 \leq k \leq m} |\operatorname{Re} \lambda_k|}{\min_{1 \leq k \leq m} |\operatorname{Re} \lambda_k|}.$$

Если матрица A не постоянна и $\lambda_k = \lambda_k(t)$, то вводят понятие **числа жесткости системы ОДУ на временном интервале**. В этом случае должно быть велико значение $\sup_{t \in (0, T)} S(t)$.

Подобное определение можно ввести и для нелинейных систем, рассмотрев их локальную линеаризацию. Отметим, что существуют и другие определения жестких систем и жесткости.

7.5.3. Решение жестких систем

Понятие условной устойчивости показывает, что порядок аппроксимации p метода решения задачи Коши не единственный показатель его эффективности. Важна также возможность изменять шаг τ в широком диапазоне без потери устойчивости.

Рассмотрим ***тестовое уравнение Далквиста***

$$u' = \lambda u,$$

где λ — параметр (комплексное число). Такой случай соответствует произвольной матрице A , имеющей, вообще говоря, комплексные собственные значения. Применение метода решения задачи Коши к уравнению Далквиста дает представление об устойчивости метода в линейном приближении правой части ОДУ.

Рассмотрим линейный многошаговый метод

$$\frac{1}{\tau} \sum_{k=0}^m a_k y_{n-k} = \sum_{k=0}^m b_k f_{n-k}.$$

Применительно к уравнению Далквиста этот метод сводится к решению следующей системы алгебраических уравнений:

$$\sum_{k=0}^m (a_k - \lambda \tau b_k) y_{n-k} = 0.$$

Введем обозначение: $\lambda \tau = \mu \in \mathbb{C}$.

Решение полученного разностного уравнения будем искать в виде $y_n = q^n$. Запишем характеристическое уравнение для данного метода:

$$\sum_{k=0}^m (a_k - \mu b_k) q^{m-k} = 0.$$

Это уравнение отличается от характеристического уравнения линейного m -шагового разностного метода (см. 7.4.4).

Определение. ***Областью устойчивости метода решения задачи Коши*** называется множество точек $\mu = \lambda \tau$ комплексной плоскости, для которых данный метод применительно к уравнению $u' = \lambda u$ устойчив. Иными словами, должно быть справедливо неравенство $|y_{n+1}| \leq |y_n|$, что выполняется, если корни характеристического уравнения лежат внутри единичного круга $|\mu| \leq 1$, а на его границе нет кратных корней.

Пример 7.3. Рассмотрим явный метод Эйлера применительно к тестовому уравнению Далквиста. В используемых обозначениях соответствующее разностное уравнение имеет вид $y_{n+1} = y_n(1 + \mu)$. Отсюда получим уравнение для области устойчивости $|1 + \mu| \leq 1$, или, полагая $\mu = \mu_x + i\mu_y$,

$$(1 + \mu_x)^2 + \mu_y^2 \leq 1.$$

Область устойчивости представляет собой круг единичного радиуса с центром в точке $(-1; 0)$ (рис. 7.1, а). •

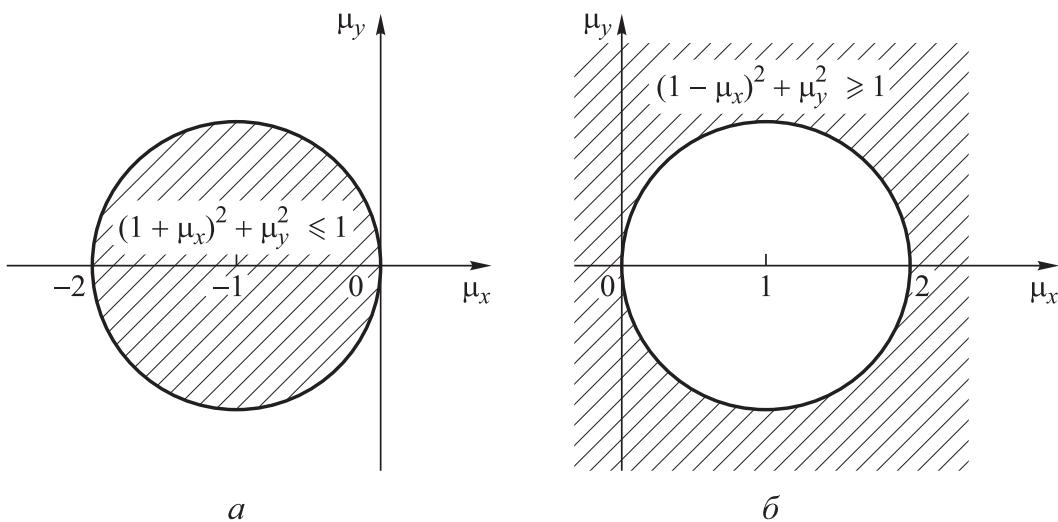


Рис. 7.1. Область устойчивости метода Эйлера:
а — явного; б — неявного

Пример 7.4. Использование неявного метода Эйлера приводит к разностному уравнению

$$y_{n+1} = y_n(1 - \mu)^{-1}.$$

Отсюда получим неравенство, описывающее область устойчивости

$$|(1 - \mu)^{-1}| \leq 1,$$

или, полагая $\mu = \mu_x + i\mu_y$,

$$(1 - \mu_x)^2 + \mu_y^2 \geq 1.$$

Область устойчивости представляет собой область вне единичного круга с центром в точке $(1; 0)$ (рис. 7.1, б). •

Определение. *Разностный метод* называется **A -устойчивым**, если область его устойчивости содержит левую полу-плоскость $\operatorname{Re} \mu < 0$.

Определение. *Разностный метод* называется **$A(\alpha)$ -устойчивым**, если существует угол $\alpha \in (0, \pi/2]$ такой, что область его устойчивости содержит сектор комплексной плоскости переменной μ , определяемый неравенством $|\arg(-\mu)| < \alpha$.

В рамках этого определения A -устойчивость есть $A(\pi/2)$ -устойчивость. Неявный метод Эйлера A -устойчив.

Определение. *Системы ОДУ*, для которых $\operatorname{Re} \lambda < 0$, называются **асимптотически устойчивыми**. Для таких систем A -устойчивые методы устойчивы для любых $\tau > 0$, что означает **безусловную устойчивость**.

Пример 7.5. Рассмотрим симметричную схему:

$$\frac{y_{n+1} - y_n}{\tau} = \lambda \frac{y_n + y_{n+1}}{2},$$

или

$$y_{n+1} = y_n \frac{1 + \mu/2}{1 - \mu/2}.$$

Отсюда получаем неравенство для определения области устойчивости:

$$\left(1 + \frac{\mu_x}{2}\right)^2 + \left(\frac{\mu_y}{2}\right)^2 \leq \left(1 - \frac{\mu_x}{2}\right)^2 + \left(\frac{\mu_y}{2}\right)^2,$$

откуда следует, что $\mu_x \leq 0$, т. е. симметричная схема A -устойчива. •

Отметим, что среди линейных явных m -шаговых разностных методов нет A -устойчивых. Покажем это. Из характеристического уравнения в силу равенства $b_0 = 0$ получаем, что для любого q справедливо равенство

$$\mu = \sum_{k=0}^m a_k q^{m-k} / \sum_{k=1}^m b_k q^{m-k}.$$

Таким образом, при больших по модулю значениях q параметр μ возрастает линейно как $(a_0/b_1)q$, если $b_1 \neq 0$, либо как более высокая степень q , если $b_1 = 0$. Следовательно, для любого достаточно большого μ найдется q из левой полуплоскости, в том числе с $|q| > 1$, для которого справедливо характеристическое уравнение. В результате A -устойчивость не имеет места.

Точно так же доказано, что ни для какого α не существует явного $A(\alpha)$ -устойчивого линейного многошагового метода. Поэтому для решения жестких систем часто используют методы Гира — полностью неявные многошаговые разностные методы высокого порядка аппроксимации, т. е.

$$\frac{1}{\tau} \sum_{i=0}^m a_i y_{n-i} = f(t_n, y_n).$$

Эти методы полностью неявные, поскольку $b_0 = 1$, $b_i = 0$, $i = \overline{1, m}$.

Пример 7.6. Рассмотрим метод Гира третьего порядка ($m = 3$) и найдем область его устойчивости. Для тестового уравнения Далквиста получаем

$$\frac{11}{6}y_n - 3y_{n-1} + \frac{3}{2}y_{n-2} - \frac{1}{3}y_{n-3} = \lambda\tau y_n = \mu y_n.$$

Найдем его решение в виде $y_n = q^n$. После подстановки и сокращения на q^{n-3} получаем характеристическое уравнение

$$\frac{11}{6}q^3 - 3q^2 + \frac{3}{2}q - \frac{1}{3} = \mu q^3.$$

Записать формально аналитическое решение кубического уравнения еще можно. Однако найти из его решения область устойчивости метода — более сложная задача. Поэтому поступим иначе. Найдем границу области устойчивости, после чего определим области устойчивости и неустойчивости методом пробных точек. Граница области устойчивости соответствует таким q , что $|q| = 1$. Для них $q = e^{i\varphi}$, где i — мнимая единица; φ — аргумент комплексного числа (действительное число).

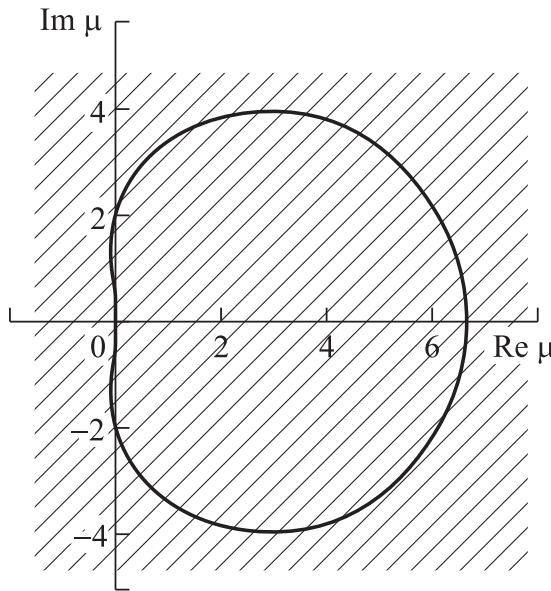


Рис. 7.2. Область устойчивости трехшагового метода Гира

Таким методом получаем параметрическое задание границы области устойчивости в виде

$$\frac{11}{6} - 3e^{-i\varphi} + \frac{3}{2}e^{-2i\varphi} - \frac{1}{3}e^{-3i\varphi} = \mu,$$

где $\varphi \in [0; 2\pi]$. Соответствующая кривая изображена на рис. 7.2. Выбрав $\mu = -35/3$, получим уравнение для определения q :

$$\frac{27}{2}q^3 - 3q^2 + \frac{3}{2}q - \frac{1}{3} = 0,$$

или

$$\frac{27}{2}q^2 \left(q - \frac{2}{9} \right) + \frac{3}{2} \left(q - \frac{2}{9} \right) = 0.$$

Корни этого уравнения:

$$q_1 = \frac{2}{9}; \quad q_{2,3} = \pm \frac{i}{3}.$$

Модули полученных корней не превышают единицы, а следовательно, область устойчивости находится вне области, граница которой определена построенной кривой. Как видно, метод не является A -устойчивым, однако он $A(\alpha)$ -устойчив («почти A -устойчив») с $\alpha \approx 19\pi/40$. •

Пример 7.7. Построим область устойчивости двухстадийного метода Рунге — Кутты второго порядка

$$\begin{aligned} k_1 &= f(t_n, y_n); \\ k_2 &= f\left(t_n + \frac{\tau}{2}, y_n + \frac{\tau}{2}k_1\right); \\ y_{n+1} &= y_n + \tau k_2. \end{aligned}$$

Запишем метод применительно к уравнению Далквиста $u' = \lambda u$:

$$k_1 = \lambda y_n; \quad k_2 = \lambda \left(y_n + \frac{\tau}{2} \lambda y_n\right); \quad y_{n+1} = y_n \left(1 + \tau \lambda + \frac{(\tau \lambda)^2}{2}\right).$$

Для устойчивости метода необходимо, чтобы $|R(\mu)| \leq 1$, где $R(\mu)$ — функция устойчивости:

$$R(\mu) = 1 + \mu + \frac{\mu^2}{2}, \quad \mu \in \mathbb{C}.$$

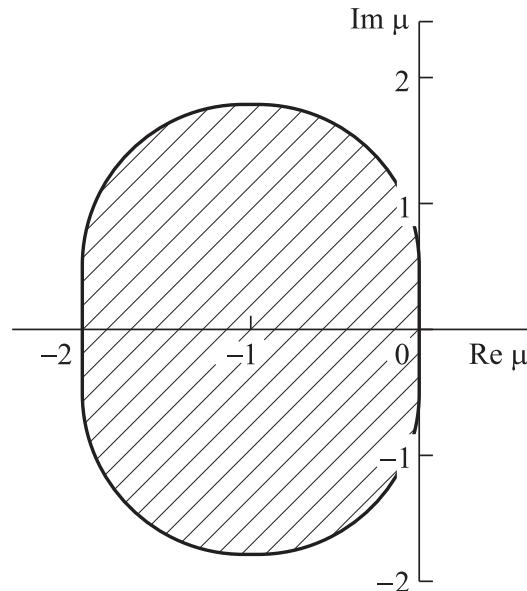


Рис. 7.3. Область устойчивости двухстадийного метода Рунге — Кутты второго порядка

Граница области устойчивости метода $\left|1 + \mu + \frac{\mu^2}{2}\right| = 1$ изображена на рис. 7.3. Очевидно, что область устойчивости лежит внутри границы: $R(-1) = 1/2$, $R(1) = 5/2$. •

Вопросы и задания

1. Сформулируйте постановку и условия разрешимости задачи Коши для ОДУ.
2. Запишите эквивалентную интегральную формулировку задачи Коши для ОДУ.
3. Запишите явный и неявный методы Эйлера численного решения задачи Коши для ОДУ.
4. Запишите симметричную схему численного решения задачи Коши для ОДУ. Какая квадратурная формула лежит в ее основе?
5. Дайте определение методов Рунге — Кутты численного решения задачи Коши для ОДУ.
6. Как связаны квадратурные формулы и формулы промежуточных стадий и приближенного решения в методах Рунге — Кутты численного решения задачи Коши для ОДУ?
7. Запишите однопараметрическое семейство двухстадийных методов Рунге — Кутты численного решения задачи Коши для ОДУ. Какие из этих методов имеют второй порядок?
8. Что такое таблицы Бутчера? Приведите примеры методов Рунге — Кутты численного решения задачи Коши для ОДУ и их таблицы Бутчера.
9. Сформулируйте условия сходимости методов Рунге — Кутты в общем случае.
10. Что называется порядком точности численного метода задачи Коши? Что называется порядком аппроксимации численного метода решения задачи Коши? Как связаны между собой порядок точности и порядок аппроксимации?

11. Как оценить погрешность численного решения ОДУ на практике? Зачем и как управлять длиной шага в процессе вычислений? Для чего используют правило Рунге и вложенные методы Рунге — Кутты?
12. Какие методы решения задачи Коши для ОДУ называют линейными многошаговыми разностными методами? Каким порядком точности и аппроксимации они обладают?
13. Какие методы относятся к семействам методов Адамса и методов Гира? Приведите примеры методов.
14. Приведите примеры построения методов Адамса и методов Гира с использованием алгоритмов интерполяции.
15. Какова погрешность аппроксимации линейного многошагового метода и что такое условия порядка?
16. Приведите примеры построения методов Адамса и методов Гира на основе условий порядка.
17. Какой метод решения задачи Коши для ОДУ называется устойчивым? Какие виды устойчивости вы знаете?
18. Что называется областью устойчивости метода решения задачи Коши для ОДУ? Какой метод называется нуль-устойчивым, а какой A -устойчивым?
19. Приведите примеры исследования устойчивости методов Адамса и методов Гира.
20. Какая система ОДУ называется жесткой? Что такое число жесткости?
21. Приведите определения условной и абсолютной устойчивости разностного метода и примеры, иллюстрирующие эти определения.

Библиографические комментарии

Теория численных методов решения задач Коши — хорошо разработанный раздел прикладной математики. Понятия, используемые в данной главе, введены в учебнике [68]. Основы теории ОДУ можно найти, например, в работе [2]. Особо отметим двухтомник [81, 82], в котором наиболее полно изложены методы численного решения ОДУ.

Подробности рассмотренных алгоритмов, а также другие способы решения задачи Коши можно найти, например, в работах [3, 8–11, 15, 23, 26, 27, 35, 41, 42, 45, 51] и различных руководствах по численным методам.

Классические и новые подходы к решению жестких задач представлены в работах [35, 51, 80].

В последнее время в связи с появлением современных пакетов программ для решения больших жестких систем ОДУ новую жизнь обрел метод прямых, предназначенный для решения эволюционных уравнений и систем таких уравнений с частными производными. При использовании метода прямых проводят дискретизацию уравнения по пространственным переменным, в результате чего записывают задачу Коши для системы ОДУ большой размерности. Эта система, как правило, оказывается жесткой. Подробное описание данного метода можно найти в работах [15, 27, 49] и другой литературе, указанной в библиографии.

8. РЕШЕНИЕ КРАЕВЫХ ЗАДАЧ ДЛЯ СИСТЕМ ОБЫКНОВЕННЫХ ДИФФЕРЕНЦИАЛЬНЫХ УРАВНЕНИЙ

Изложены простейшие методы решения краевых задач для ОДУ. Представлен метод стрельбы для одного и нескольких уравнений. На примере простейшей краевой задачи для уравнения второго порядка (линейного и нелинейного) описан разностный метод с доказательством сходимости, обоснована применимость метода Ньютона в нелинейном случае. Приведен пример точной разностной схемы, в данном случае — схемы с экспоненциальной подгонкой. Описаны методы Ритца и Галёркина.

8.1. Постановка задачи. Метод стрельбы

Пусть на отрезке $[a, b]$ задана система n ОДУ первого порядка в нормальной форме:

$$u' = f(x, u), \quad a \leq x \leq b,$$

где u и $f(x, u)$ — векторные функции.

Для определения единственного решения такой задачи необходимо задать n дополнительных условий. Если они заданы более чем в одной точке, то получаем так называемую краевую задачу. Рассмотрим случай, когда дополнительные условия заданы в двух точках и имеют вид

$$\begin{aligned} \varphi_k(u_1(a), \dots, u_n(a)) &= 0, \quad k = \overline{1, m}; \\ \varphi_k(u_1(b), \dots, u_n(b)) &= 0, \quad k = \overline{m + 1, n}. \end{aligned}$$

Здесь φ_k — заданные функции, при этом уравнения в точках a и b напрямую не связаны между собой.

Для решения этой задачи применим **метод стрельбы**. Алгоритм этого метода основан на сведении краевой задачи к задаче Коши для такой же системы ОДУ. Рассмотрим два случая: $n = 2$ и $n \geq 3$.

Случай $n = 2$. Опишем алгоритм метода стрельбы на примере решения системы двух ОДУ первого порядка ($n = 2$) и с двумя краевыми условиями, одно из которых задано в точке $x = a$, а другое — в точке $x = b$:

$$\begin{aligned}\varphi_1(u_1(a), u_2(a)) &= 0; \\ \varphi_2(u_1(b), u_2(b)) &= 0.\end{aligned}$$

Введем обозначения: y_1 и y_2 — численное решение, приближающее (аппроксимирующее) решение краевой задачи u_1 и u_2 соответственно. Зададим одно из граничных значений неизвестной функции равным η , например: $y_2(a) = \eta$, тогда левое краевое условие можно рассматривать как уравнение $\varphi_1(y_1(a), \eta) = 0$ относительно $y_1(a)$. Очевидно, что описываемый алгоритм имеет смысл только в том случае, если это уравнение разрешимо и дает результат $y_1(a, \eta) = \xi(\eta)$.

Примем $y_2(a) = \eta$ и $y_1(a) = \xi(\eta)$ в качестве начальных условий и поставим задачу Коши для системы ОДУ $y' = f(t, y)$, где $y = (y_1, y_2)^T$. Решим ее любым численным методом, например методом Рунге — Кутты (см. 7.3). Получим решение $y_1(x)$ и $y_2(x)$ для данного значения η . Варьирование η позволяет получить $y_1(x, \eta)$ и $y_2(x, \eta)$, зависящие от η как от параметра. Найденное решение необязательно удовлетворяет правому краевому условию:

$$\varphi_2(y_1(b, \eta), y_2(b, \eta)) = \psi(\eta) \neq 0.$$

Необходимо подобрать значение η (с некоторой точностью) так, чтобы $\psi(\eta) \approx 0$. Таким образом, решение краевой задачи сводится к поиску корней алгебраического уравнения $\psi(\eta) = 0$.

Соответствующие методы рассмотрены в главе 5, однако вычисление одного значения функции $\psi(\eta)$ требует численного решения задачи Коши для системы ОДУ.

Один из наиболее простых способов решения уравнения $\psi(\eta) = 0$ — описанный в 5.1.2 *метод деления пополам* некоторого отрезка $[\eta^1, \eta^2]$ такого, что $\psi(\eta^1) < 0$, а $\psi(\eta^2) > 0$ (знаки ψ могут быть обратными). Сначала делают пробные «выстрелы» — расчеты с произвольно выбранными значениями η^k и η^{k+1} , проводимые до тех пор, пока среди значений $\psi(\eta^k)$ не найдется пары значений разных знаков, условно этот вариант метода можно назвать «*перелет — недолет*». Этую пару η^k и η^{k+1} используют в качестве начальной для метода вилки, и применяют его до получения требуемой точности $\psi((\eta^k + \eta^{k+1})/2) \approx 0$. Для ускорения сходимости можно сконструировать и реализовать варианты методов секущих, парабол или Ньютона.

Пример 8.1. Рассмотрим краевую задачу для уравнения второго порядка:

$$\begin{aligned} u'' &= f(x, u); \\ u(a) &= \mu_1; \quad u(b) = \mu_2. \end{aligned}$$

Введем пару неизвестных функций $u_1 = u$, $u_2 = u'$. Нормализованная система для новых функций u_1 , u_2 , соответствующая уравнению второго порядка, имеет вид

$$\begin{aligned} u'_1 &= u_2; \\ u'_2 &= f(x, u_1); \\ u_1(a) &= \mu_1; \quad u_1(b) = \mu_2. \end{aligned}$$

Пусть y_1 , y_2 — численное решение задачи, приближающее искомые функции u_1 , u_2 . Границные условия для u_2 отсутствуют; положим $y_2(a) = \eta$. Поскольку $u_2(a) = u'_1(a)$, то η есть тангенс угла наклона касательной к кривой $y_1(x, \eta)$ в точке $x = a$.

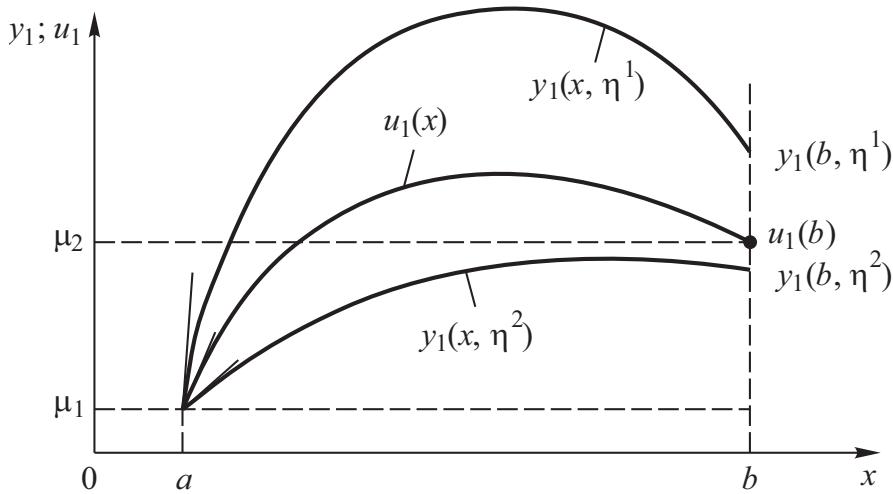


Рис. 8.1. Подбор параметра η в методе стрельбы

Выбор параметра η схематично показан на рис. 8.1. Для каждого значения η ищется приближенное решение задачи Коши $y_1(x, \eta)$, $y_2(x, \eta)$ для системы ОДУ с начальными данными $u_1(a) = \mu_1$, $u_2(a) = \eta$. Найденное решение — это «выстрел», который либо «попал» в значение граничного условия $u_1(b) = \mu_2$, либо «не попал». •

Метод стрельбы особенно легко применять для решения линейных задач, так как в этом случае решение линейным образом зависит от η .

Рассмотрим следующую задачу:

$$\begin{aligned} u'_1 &= a_{11}(x)u_1 + a_{12}(x)u_2 + f_1(x); \\ u'_2 &= a_{21}(x)u_1 + a_{22}(x)u_2 + f_2(x) \end{aligned}$$

с краевыми условиями

$$\begin{aligned} \alpha_1 u_1(a) + \alpha_2 u_2(a) &= \gamma_1; \\ \beta_1 u_1(b) + \beta_2 u_2(b) &= \gamma_2. \end{aligned}$$

Предположим, что $u_2(a) = \eta$, $\alpha_1 \neq 0$, y_1 , y_2 — решения задачи Коши с начальными условиями $\alpha_1 u_1(a) + \alpha_2 u_2(a) = \gamma_1$, $u_2(a) = \eta$. При $\eta = 0$ найдем y_1^0 , y_2^0 , а при $\eta = 1$ — y_1^1 , y_2^1 .

Поскольку рассматриваемая система ОДУ линейная, то решение $y_1(x, \eta)$, $y_2(x, \eta)$ линейно относительно η . Поэтому решение можно записать в виде

$$y_1(x, \eta) = y_1^0(x) + (y_1^1(x) - y_1^0(x))\eta;$$

$$y_2(x, \eta) = y_2^0(x) + (y_2^1(x) - y_2^0(x))\eta.$$

Тогда получаем, что на правой границе должно быть выполнено условие

$$\begin{aligned} \beta_1 \left[y_1^0(b) + (y_1^1(b) - y_1^0(b))\eta \right] + \\ + \beta_2 \left[y_2^0(b) + (y_2^1(b) - y_2^0(b))\eta \right] = \gamma_2, \end{aligned}$$

т. е. η — решение данного линейного уравнения:

$$\eta = \frac{\gamma_2 - \beta_1 y_1^0(b) - \beta_2 y_2^0(b)}{\beta_1 (y_1^1(b) - y_1^0(b)) + \beta_2 (y_2^1(b) - y_2^0(b))}.$$

Следовательно, для отыскания решения краевой задачи необходимо решить всего две задачи Коши при $\eta = 0$ и $\eta = 1$, найти η из граничного условия и получить y_1 , y_2 .

При конкретной реализации возникает вопрос о том, с какой границы (a или b) нужно стартовать при стрельбе. Может возникнуть ситуация, при которой краевая задача для исходной системы устойчива, а задача Коши — нет. Тогда решение задачи Коши будет иметь низкую точность. Необходимо соответствующим образом скорректировать метод решения начальной задачи.

Пример 8.2. Рассмотрим следующую краевую задачу:

$$u'' - u = 0, \quad u = u(x), \quad 0 < x < 1;$$

$$u(0) = 1; \quad u(1) = 2.$$

Ее точное решение имеет вид

$$u(x) = (2 \operatorname{sh} x - \operatorname{sh}(x - 1)) / \operatorname{sh} 1.$$

Это решение содержит две экспоненты: возрастающую и убывающую. Следовательно, задача Коши для данного уравнения будет неустойчивой вне зависимости от того, на каком конце отрезка $[0; 1]$ заданы так называемые начальные условия. В то же время численное решение данной задачи именно методами решения краевых задач сложности не представляет. ●

Случай $n \geq 3$. Постановка задачи включает m краевых условий в точке a , где $1 \leq m \leq n - 1$, и $n - m$ условий в точке b . При решении задачи методом стрельбы необходимо задать $n - m$ параметров $\eta_1, \eta_2, \dots, \eta_{n-m}$, определяющих значения неизвестных функций $u_{m+1}(a), \dots, u_n(a)$. Тогда значения $u_1(a), u_2(a), \dots, u_m(a)$ являются решениями системы m уравнений

$$\varphi_k(u_1(a), \dots, u_n(a)) = 0, \quad k = \overline{1, m}.$$

В результате получаем задачу Коши. Решив эту задачу, вычисляем $n - m$ значений

$$\varphi_k(y_1(b), \dots, y_n(b)) = \psi_k(\eta_1, \dots, \eta_{n-m}), \quad k = \overline{m + 1, n},$$

отличных, вообще говоря, от нуля. Необходимо подобрать $\eta_1, \eta_2, \dots, \eta_{n-m}$ так, чтобы все $\psi_k = 0, k = \overline{m + 1, n}$.

В нелинейном случае, когда вид функций ψ_k неизвестен, подбор параметров $\eta_1, \eta_2, \dots, \eta_{n-m}$ очень сложен. Поэтому в нелинейном случае метод стрельбы при $n - m > 1$ практически не используется.

В линейном случае изложенный метод решения задачи с $n = 2$ легко обобщается и на задачи с $n \geq 3$. При этом необходимо решить $n - m + 1$ раз задачу Коши, а также еще одну СЛАУ для определения $\eta_1, \eta_2, \dots, \eta_{n-m}$.

Замечание 8.1. В настоящее время метод стрельбы сравнительно редко применяют для нахождения решений собственно краевых задач. Однако методы этого класса довольно часто

используют при решении задач на собственные значения для отыскания параметра, при котором решение однородной задачи нетривиально.

8.2. Конечно-разностные методы

8.2.1. Линейная краевая задача второго порядка

Рассмотрим конечно-разностные методы решения краевой задачи для ОДУ второго порядка:

$$u'' - p(x)u = f(x), \quad a < x < b;$$

$$u(a) = \alpha; \quad u(b) = \beta.$$

Разностная сетка и аппроксимация. На отрезке $[a, b]$ введем сетку (для простоты изложения возьмем сетку с постоянным шагом h):

$$\omega_h = \left\{ x_i = a + ih, \quad i = \overline{0, N}; \quad h = \frac{b - a}{N} \right\}.$$

Вместо непрерывной функции (или вектор-функции) будем искать приближенное решение лишь в точках сетки, т.е. сеточную функцию.

В каждой точке сетки производные в основном уравнении заменим разностными аналогами. Для построения разностных аналогов можно использовать тот же подход, который применялся для получения аппроксимации производных с различным порядком точности (см. 6.7).

Стандартная схема. Введя сетку и заменив в уравнении второго порядка вторую производную разностным соотношением, получим СЛАУ с трехдиагональной матрицей:

$$\frac{1}{h^2}(y_{i+1} - 2y_i + y_{i-1}) - p(x_i)y_i = f(x_i), \quad i = \overline{1, N-1};$$

$$y_0 = y(x_0) = \alpha; \quad y_N = y(x_N) = \beta.$$

Эту систему уравнений можно решать методом прогонки. Если $p(x) > 0$, то алгоритм прогонки устойчив и проблем с ее реализацией не возникает, поскольку в системе есть строгое диагональное преобладание:

$$\frac{2}{h^2} + p(x_i) > \frac{1}{h^2} + \frac{1}{h^2}.$$

Теорема 8.1. Если $f, p \in C^2(a, b)$ и $p(x) > 0$ при $x \in [a, b]$, то разностное решение равномерно сходится к точному со скоростью $O(h^2)$.

◀ Будем пользоваться теми же терминами, что и при исследовании методов для задачи Коши. Для погрешности $z_i = y_i - u_i$ имеем задачу

$$\begin{aligned} \frac{1}{h^2}(z_{i+1} - 2z_i + z_{i-1}) - p_i z_i = \\ = f(x_i) - \frac{1}{h^2}(u_{i+1} - 2u_i + u_{i-1}) + p_i u_i = \psi_h^{(1)} + \psi_h^{(2)}, \end{aligned}$$

где $\psi_h^{(2)} = 0$, а невязка на точном решении задачи

$$\begin{aligned} \psi_h^{(1)} = f(x_i) - \frac{1}{h^2}(u_{i+1} - 2u_i + u_{i-1}) + p_i u_i = \\ = u_i'' - \frac{1}{h^2} \left(u_i'' h^2 + \frac{1}{12} u_i^{(4)} h^4 \right) = -\frac{1}{12} u_i^{(4)} h^2. \end{aligned}$$

Существование и ограниченность $u^{(4)}$ гарантируются условиями теоремы ($f, p \in C^2(a, b)$). Имеем $|u^{(4)}| \leq M_4$, $z_0 = z_N = 0$. Тогда z_i в какой-то внутренней точке сетки достигает своего максимума (по модулю). Пусть это будет точка i_0 . Тогда для этой точки

$$(2 + p_{i_0} h^2) z_{i_0} = z_{i_0+1} + z_{i_0-1} - h^2 \psi_h^{(1)},$$

откуда

$$|z_{i_0}| \leq \frac{h^2}{12} \frac{u^{(4)}}{p_{i_0}} \Rightarrow \|z\|_C \leq \frac{h^2}{12} \max_{x, x^* \in (a, b)} \left| \frac{u^{(4)}(x)}{p(x^*)} \right|,$$

т. е. $\|z\|_C = O(h^2)$. ►

Специализированная схема. Разберем на примере рассмотренной задачи алгоритм построения специализированных схем, позволяющих воспроизвести типичное для данной задачи точное решение. Потребуем от конструируемой разностной схемы, чтобы она была точна на решениях исходной задачи в случае $p = p(x_i) = \text{const}$, $f = f(x_i) = \text{const}$. Ограничимся, как и ранее, равномерной сеткой.

В рассматриваемом варианте точное решение можно записать в форме

$$u(x) = -\frac{f(x_i)}{p(x_i)} + A \exp\left(\sqrt{p(x_i)}x\right) + B \exp\left(-\sqrt{p(x_i)}x\right).$$

Возьмем разностную схему вида

$$\frac{1}{h^2}(y_{i+1} - 2y_i + y_{i-1}) - a_i y_i = \frac{a_i}{p(x_i)} f(x_i), \quad i = \overline{1, N-1},$$

и подберем параметр a_i так, чтобы точное решение исходной дифференциальной задачи было также точным решением системы алгебраических уравнений. Указанному требованию удовлетворяет параметр

$$a_i = \frac{4}{h^2} [\operatorname{sh}(0,5\sqrt{p(x_i)}h)]^2.$$

По построению схема является точной на решениях данного вида. Такие схемы часто называют точными разностными схемами.

Конкретная схема, построенная для данной задачи, иногда называется **схемой с экспоненциальной подгонкой**. Особенно успешно по сравнению с традиционными такие схемы применяются в случае больших значений параметра p , т. е. для решения задач с малым параметром при старшей производной. Такие задачи обладают свойством, усложняющим численное решение: размеры области, в которых происходит резкое изменение решения, много меньше рассматриваемого пространственного

(или временного) участка. По этой причине их также иногда называют жесткими краевыми задачами.

Отметим, что при малом p , таком что $|0,5\sqrt{p(x_i)}h| \ll 1$, получаем $a_i \approx p(x_i)$, т. е. стандартную схему решения линейной краевой задачи второго порядка.

8.2.2. Нелинейные задачи

Рассмотрим простейшую нелинейную краевую задачу:

$$\begin{aligned} u'' &= f(x, u), \quad x \in (a, b); \\ u(a) &= \alpha; \quad u(b) = \beta. \end{aligned}$$

Пусть $f'_u \geq m > 0$.

Для решения задачи запишем разностную схему того же вида, что и в линейном случае (см. 8.2.1):

$$\begin{aligned} \frac{1}{h^2}(y_{i+1} - 2y_i + y_{i-1}) &= f(x_i, y_i), \quad i = \overline{1, N-1}; \\ y_0 &= \alpha, \quad y_N = \beta. \end{aligned}$$

Докажем сходимость приближенного решения к точному в предположении существования обоих решений и выполнения условий

$$|u^{(4)}| \leq M_4; \quad f'_u \geq m > 0.$$

Аналогично линейному случаю получим

$$\begin{aligned} \psi_h^{(1)} &= f(x_i, u_i) - \frac{1}{h^2}(u_{i+1} - 2u_i + u_{i-1}) = -\frac{1}{12}u_i^{(4)}h^2; \\ \psi_h^{(2)} &= f(x_i, u_i + z_i) - f(x_i, u_i) = f'_u(x_i, u_i + \theta z_i)z_i, \end{aligned}$$

где $\theta \in [0; 1]$.

Отсюда уравнения для погрешности z_i имеют вид

$$\begin{aligned} \frac{1}{h^2}(z_{i+1} - 2z_i + z_{i-1}) - f'_u(x_i, y_i + \theta z_i)z_i &= \psi_h^{(1)}; \\ z_0 &= z_N = 0. \end{aligned}$$

В результате точно так же, как и в линейном случае, получим оценку погрешности

$$\|z\|_C \leq \frac{h^2}{12} \frac{M_4}{m},$$

т. е. $\|z\|_C = O(h^2)$.

Для нахождения решения системы нелинейных алгебраических уравнений необходимо использовать какой-либо итерационный метод. Наиболее часто применяют метод Ньютона. Рассмотрим его подробнее.

Пусть y_i^s — приближенное решение разностной схемы на s -й итерации. Линеаризация правой части этой схемы дает

$$f(x_i, y_i^{s+1}) \approx f(x_i, y_i^s) + f'_u(x_i, y_i^s)(y_i^{s+1} - y_i^s).$$

Пусть $\delta y_i^s = y_i - y_i^s$. Тогда из уравнений итерационного метода

$$\frac{1}{h^2}(y_{i+1}^{s+1} - 2y_i^{s+1} + y_{i-1}^{s+1}) = f(x_i, y_i^s) + f'_u(x_i, y_i^s)(y_i^{s+1} - y_i^s)$$

и исходных нелинейных уравнений после вычитания получим

$$\begin{aligned} \frac{1}{h^2}(\delta y_{i+1}^{s+1} - 2\delta y_i^{s+1} + \delta y_{i-1}^{s+1}) - f'_u(x_i, y_i^s)\delta y_i^{s+1} &= \\ = f(x_i, y_i^s + \delta y_i^s) - f(x_i, y_i^s) - f'_u(x_i, y_i^s)\delta y_i^s &= \\ = \frac{1}{2}f''_{uu}(x_i, y_i^s + \theta\delta y_i^s)(\delta y_i^s)^2. \end{aligned}$$

Следовательно, как и в линейном случае, имеем

$$\|\delta y^{s+1}\|_C \leq \frac{1}{2} \left\| \frac{f''_{uu}}{f'_u} \right\|_C \|\delta y^s\|_C^2.$$

В результате при выборе начального приближения, близкого к точному решению разностной схемы, итерации метода Ньютона будут сходиться, и притом с квадратичной скоростью. Если такие итерации сходятся, то в силу непрерывности функции $f(x, u)$ они сходятся к точному решению исходной системы нелинейных

алгебраических уравнений. Критерий прекращения итераций может быть выбран в форме, соответствующей данному случаю.

Отметим, что наличие известного порядка сходимости позволяет применять для уточнения решения правило Рунге. Иногда этот прием может существенно сократить трудозатраты на получение решения требуемого качества.

8.3. Методы Ритца и Галёркина

Методы Ритца и Галёркина принадлежат к семейству проекционных методов. С методами данного класса ознакомимся лишь на простейшем примере.

Пусть требуется найти решение следующей задачи:

$$Au = f, \quad u = u(x), \quad a < x < b;$$

$$u(a) = \alpha; \quad u(b) = \beta.$$

Оператор A задан, $f \in F$, $D(A) \subset U$, $\text{im } A \subset F$.

Будем искать приближенное решение y_h в виде

$$u \approx y_h(x) = \varphi_0(x) + \sum_{i=1}^n c_i \varphi_i(x),$$

где $\varphi_0(x)$ — некоторая гладкая функция, удовлетворяющая граничным условиям, т. е. $\varphi_0(a) = \alpha$, $\varphi_0(b) = \beta$, $\{\varphi_i\}$ — выбранная система линейно независимых функций, полная в пространстве U , причем $\varphi \in D(A)$, $i = \overline{1, n}$, и все функции φ_i , $i = \overline{1, n}$, обращаются в нуль в краевых точках a и b .

Заданный вид приближенного решения можно интерпретировать следующим образом. Вместо точного решения операторного уравнения $Au = f$ будем искать проекцию этого решения на линейное подпространство, заданное базисом $\{\varphi_i\}_{i=1}^n$. Способ определения коэффициентов зависит от конкретного проекционного метода.

Метод Ритца. Рассмотрим функционал

$$\Phi[u] = \int_a^b (Au - f)^2 \rho dx,$$

где $\rho > 0$ — весовая функция.

Очевидно, что решение уравнения $Au = f$ обеспечивает абсолютный минимум этому функционалу. Вместе с тем абсолютный минимум функционала Φ , равный нулю, заведомо дает решение исходного уравнения $Au = f$, поскольку этот минимум соответствует таким функциям u , что $Au - f = 0$. При этом предполагаем существование точки абсолютного минимума (на функциях, удовлетворяющих заданным граничным условиям), так как функционал ограничен снизу и непрерывно зависит от Au .

Метод сведения задачи $Au = f$ к задаче минимизации $\Phi[u]$ обычно называют **методом наименьших квадратов**.

В случае линейного самосопряженного положительного оператора A , т.е. $A = A^*$, $A > 0$, можно указать и другой функционал, минимизация которого дает решение исходной задачи:

$$\Phi[u] = (u, Au) - 2(u, f).$$

Пусть $u = \bar{u} + \lambda\delta u$. Тогда

$$\Phi[u] = \Phi[\bar{u}] + \lambda^2(\delta u, A\delta u) + 2\lambda(A\bar{u} - f, \delta u).$$

Если \bar{u} таково, что $A\bar{u} = f$, то $\Phi[u] \geq \Phi[\bar{u}]$ для любых λ , δu в силу положительности оператора A , т.е. \bar{u} реализует минимум $\Phi[u]$. В то же время из условия достижения минимума при $u = \bar{u}$ имеем уравнение

$$\left. \frac{\partial \Phi}{\partial \lambda} \right|_{\lambda=0} = 0.$$

Из последнего уравнения следует, что для произвольного δu выполняется равенство $(A\bar{u} - f, \delta u) = 0$, в том числе и для $\delta u = A\bar{u} - f$, откуда $A\bar{u} - f = 0$, или $A\bar{u} = f$.

Таким образом, задача определения минимума этого функционала также эквивалентна поиску решения задачи $Au = f$.

Пусть $\Phi[u] = (u, Au) - 2(u, f)$. Будем искать решение u в виде

$$u \approx y_h(x) = \varphi_0(x) + \sum_{i=1}^n c_i \varphi_i(x).$$

Тогда

$$\begin{aligned} \Phi[y_h] &= \sum_{i=1}^n \sum_{j=1}^n c_i c_j (\varphi_i, A\varphi_j) + \\ &+ 2 \sum_{k=1}^n c_k [(\varphi_k, A\varphi_0) - (\varphi_k, f)] + (\varphi_0, A\varphi_0 - 2f) \rightarrow \min. \end{aligned}$$

Ищем коэффициенты c_i , при которых Φ на функциях данного класса достигает минимума. Условия экстремума Φ имеют вид

$$\sum_{i=1}^n c_i (\varphi_i, A\varphi_j) = -(\varphi_j, A\varphi_0 - f), \quad j = \overline{1, n}.$$

Решив полученную СЛАУ, находим приближенное решение y_h .

Описанный метод приближенного решения называют **методом Ритца**.

Метод Галёркина. Пусть $\{\varphi_i(x)\}$, $i = 1, 2, \dots$, — некоторая полная система функций. Если

$$(F, \varphi_i) = 0, \quad i = 1, 2, \dots,$$

то $F \equiv 0$ в силу полноты системы функций.

Тогда, если найти такую функцию u , что

$$(Au - f, \varphi_i) = 0, \quad i = 1, 2, \dots,$$

это означает, что u — решение исходной задачи $Au = f$. Если же такая ортогональность имеет место лишь при $i \leq n$, то,

очевидно, полученная функция u есть решение исходной задачи с точностью до линейной комбинации $\varphi_{n+1}, \varphi_{n+2}, \dots$.

Возьмем приближенное решение в виде

$$u \approx y_h(x) = \varphi_0(x) + \sum_{i=1}^n c_i \varphi_i(x)$$

и потребуем выполнения условий

$$(Ay_h - f, \varphi_i) = 0, \quad i = \overline{1, n}.$$

Получим систему n уравнений для n коэффициентов:

$$\sum_{i=1}^n c_i (A\varphi_i, \varphi_j) = -(\varphi_j, A\varphi_0 - f), \quad j = \overline{1, n}.$$

Таким образом, записаны те же уравнения, что и в случае метода Ритца. При этом требования положительности и самосопряженности оператора A не накладывали. Описанный метод называется **методом Галёркина**. Вопрос о сходимости этого метода не рассматриваем.

Выбор системы функций. Качество получаемого приближенного решения в методах Ритца и Галёркина в значительной степени зависит от выбранной системы функций (при заданном количестве n базисных функций). Часто выбирают базисную функцию

$$\varphi_0 = \alpha + \frac{\beta - \alpha}{b - a}(x - a),$$

что позволяет удовлетворять граничным условиям, однако φ_i могут быть самыми разнообразными.

Если $\varphi_i(x)$ — базисные функции метода конечных элементов, построенных в соответствии с некоторой сеткой, то будет получен вариант *метода конечных элементов*. По форме полученная СЛАУ для коэффициентов будет напоминать обычную разностную схему. Если в качестве $\varphi_i(x)$ задана система тригонометрических функций, то будут получены СЛАУ так называемого *спектрального метода*.

Вообще **методы**, основанные на проецировании исходного уравнения на некоторые последовательности подпространств $U_n \subset D(A)$ и $F_n \subset F$ и поиске решений $y_n \in U_n$, называются **проекционными**. Часто при этом решение ищется в виде

$$y_h = \varphi_0 + \sum_{i=1}^n c_i \varphi_i.$$

Если базисные функции заданы с помощью какой-либо сетки, то такие **методы** называются **проекционно-сеточными**. Отметим, что описанные приближенные методы позволяют получать функцию y_h , определенную при произвольном x , а не только в узлах сетки, так как φ_i заданы на всем отрезке $[a, b]$.

Вопросы и задания

1. Какая задача для ОДУ называется краевой?
2. Сформулируйте алгоритм решения краевой задачи для ОДУ второго порядка методом стрельбы.
3. Сформулируйте алгоритм решения краевой задачи системы двух ОДУ первого порядка методом стрельбы.
4. Как изменится алгоритм метода стрельбы при решении краевой задачи для системы более двух ОДУ первого порядка?
5. Приведите алгоритм решения краевой задачи для линейных систем ОДУ методом стрельбы.
6. Приведите алгоритм решения краевой задачи для нелинейных ОДУ на основе разностного метода.
7. Сформулируйте алгоритм решения краевой задачи для ОДУ методом Ритца.
8. Сформулируйте алгоритм решения краевой задачи для ОДУ методом Галёркина.

9. Обоснуйте выбор системы функций для решения краевой задачи для ОДУ методами Ритца или Галёркина.

Библиографические комментарии

Материал данной главы изложен практически в любой литературе соответствующей тематики. Подробности рассмотренных алгоритмов и другие способы решения краевых задач можно найти в работах [3, 8–11, 15, 23, 26, 27, 35, 41, 42, 45, 51] и различных руководствах по численным методам.

Классические и новые подходы к решению жестких задач представлены в работах [35, 51, 80]. Описание и анализ схем с экспоненциальной подгонкой можно найти в [8, 9, 80].

Особо отметим двухтомник [81, 82], в котором наиболее полно изложены методы численного решения обыкновенных дифференциальных уравнений.

Существуют методы решения уравнений в частных производных, сводящие процедуру решения исходной задачи к поиску решения некоторой краевой задачи. В учебном пособии [49] они называются поперечными методами прямых. При этом в случае, например, эволюционных задач происходит дискретизация по времени. В результате остается некоторая краевая задача для нахождения решения на данном временном слое. При наличии современных пакетов программ для решения больших жестких систем обыкновенных дифференциальных уравнений такой алгоритм может оказаться весьма эффективным.

Описание вариационных и проекционных методов можно найти в работах [26, 27, 54–56] и многих других руководствах.

Алгоритмы численного решения экстремальных задач, которые могут быть использованы при реализации вариационных методов, приведены в работе [14].

Литература

1. *Абрамович М., Стиган И.* Справочник по специальным функциям с формулами, графиками и математическими таблицами. М.: Наука, 1979. 832 с.
2. *Агафонов С.А., Герман А.Д., Муратова Т.В.* Дифференциальные уравнения. М.: Изд-во МГТУ им. Н.Э. Баумана, 1997. 336 с.
3. *Амосов А.А., Дубинский Ю.А., Копченова Н.В.* Вычислительные методы для инженеров. М.: Высш. шк., 1994. 544 с.
4. *Аристова Е.И., Завьялова Н.А., Лобанов А.И.* Практические занятия по вычислительной математике в МФТИ. Ч. I. М.: МФТИ, 2014. 243 с.
5. *Аристова Е.И., Лобанов А.И.* Практические занятия по вычислительной математике в МФТИ. Ч. II. М.: МФТИ, 2015. 310 с.
6. *Бабенко К.И.* Основы численного анализа. М.; Ижевск: НИЦ «Регулярная и хаотическая динамика», 2002. 848 с.
7. *Бабушка И., Витасек Э., Прагер М.* Численные процессы решения дифференциальных уравнений. М.: Мир, 1969. 368 с.
8. *Бахвалов Н.С.* Численные методы. М.: Наука, 1973. 632 с.
9. *Бахвалов Н.С., Жидков Н.П., Кобельков Г.М.* Численные методы. М.: Наука. Физматлит, 1978. 512 с.
10. *Березин И.С., Жидков Н.П.* Методы вычислений. Т. 1. М.: Физматгиз, 1962. 464 с.

11. *Березин И.С., Жидков Н.П.* Методы вычислений. Т. 2. М.: Физматгиз, 1960. 620 с.
12. *Боровин Г.К., Комаров М.М., Ярошевский В.С.* Ошибки-ловушки при программировании на фортране. М.: Наука. Физматлит, 1987. 144 с.
13. *Варга Р.* Функциональный анализ и теория аппроксимации в численном анализе. М.: Мир, 1974. 126 с.
14. *Васильев Ф.П.* Численные методы решения экстремальных задач. М.: Наука, 1988. 552 с.
15. *Вержбицкий В.М.* Основы численных методов. М.: Выш. шк., 2002. 840 с.
16. *Владимиров В.С.* Уравнения математической физики. М.: Наука, 1971. 512 с.
17. *Воеводин В.В.* Вычислительные основы линейной алгебры. М.: Наука, 1977. 304 с.
18. *Воеводин В.В.* Математические модели и методы в параллельных процессах. М.: Наука. Физматлит, 1986. 296 с.
19. *Воеводин В.В., Кузнецов Ю.А.* Матрицы и вычисления. М.: Наука, 1984. 320 с.
20. *Волков Е.А.* Численные методы. М.: Наука, 1982. 254 с.
21. *Галанин М.П., Савенков Е.Б.* Методы численного анализа математических моделей. М.: Изд-во МГТУ им. Н.Э. Баумана, 2010. 591 с.
22. *Гантмахер Ф.Р.* Теория матриц. М.: Наука. Физматлит, 1967. 576 с.
23. *Годунов С.К., Рябенький В.С.* Разностные схемы. М.: Наука, 1977. 440 с.
24. *Голуб Дж., Ван Лоун Ч.* Матричные вычисления. М.: Мир, 1999. 548 с.

25. Гулин А.В., Мажорова О.С., Морозова В.А. Введение в численные методы в задачах и упражнениях. М.: АРГАМАК-МЕДИА: ИНФРА-М, 2014. 368 с.
26. Демидович Б.П., Марон И.А. Основы вычислительной математики. М.: Наука, 1970. 664 с.
27. Демидович Б.П., Марон И.А., Шувалова Э.З. Численные методы анализа. М.: Наука, 1962. 367 с.
28. Деммель Дж. Вычислительная линейная алгебра. Теория и приложения. М.: Мир, 2001. 430 с.
29. Деннис Дж., Шнабель Р. Численные методы безусловной оптимизации и решения нелинейных уравнений. М.: Мир, 1998. 440 с.
30. Джордж А., Лю Дж. Численное решение больших разреженных систем уравнений. М.: Мир, 1984. 333 с.
31. Зарубин В.С. Математическое моделирование в технике / под ред. В.С. Зарубина, А.П. Крищенко. М.: Изд-во МГТУ им. Н.Э. Баумана, 2001. 496 с.
32. Иванов В.В. Методы вычислений на ЭВМ. Киев: Наукова думка, 1986. 584 с.
33. Икрамов Х.Д. Численные методы для симметричных линейных систем. М.: Наука, 1988. 159 с.
34. Ильин В.П. Методы и технологии конечных элементов. Новосибирск: Изд-во ИВМ и МГ СО РАН, 2007. 371 с.
35. Калиткин Н.Н. Численные методы. М.: Наука, 1978. 512 с.
36. Калиткин Н.Н., Альшин А.Б., Альшина Е.А., Рогов Б.В. Вычисления на квазиволновых сетках. М.: Физматлит, 2005. 224 с.
37. Калиткин Н.Н., Альшина Е.А. Численный анализ. Кн. 1. М.: Издательский центр «Академия», 2015. 304 с.

38. Калиткин Н.Н., Корякин П.В. Методы математической физики. Кн. 2. М.: Издательский центр «Академия», 2015. 304 с.
39. Канатников А.Н., Крищенко А.П. Линейная алгебра / под ред. В.С. Зарубина, А.П. Крищенко. М.: Изд-во МГТУ им. Н.Э. Баумана, 1999. 336 с.
40. Канатников А.Н., Крищенко А.П., Четвериков В.Н. Дифференциальное исчисление функций многих переменных / под ред. В.С. Зарубина, А.П. Крищенко. М.: Изд-во МГТУ им. Н.Э. Баумана, 2000. 456 с.
41. Канторович Л.В., Крылов В.И. Приближенные методы высшего анализа. М.; Л.: Физматгиз, 1962. 708 с.
42. Кахранер Д., Моулер К., Нэш С. Численные методы и программное обеспечение. М.: Мир, 1998. 575 с.
43. Коллатц Л. Функциональный анализ и вычислительная математика. М.: Мир, 1969. 448 с.
44. Коллатц Л. Задачи на собственные значения. М.: Наука. 1968. 503 с.
45. Копченова Н.В., Марон Н.А. Вычислительная математика в примерах и задачах. М.: Наука. Физматлит, 1972. 368 с.
46. Коробов Н.М. Теоретико-числовые методы в приближенном анализе. М.: Физматгиз, 1963. 224 с.
47. Костомаров Д.П., Фаворский А.П. Вводные лекции по численным методам. М.: Логос, 2004. 184 с.
48. Красносельский М.А., Вайникко Г.М., Забрейко П.П., Рутицкий Я.Б., Стеценко В.Я. Приближенное решение операторных уравнений. М.: Наука. Физматлит, 1969. 456 с.
49. Крылов В.И., Бобков В.В., Монастырный П.И. Вычислительные методы. Т. 1. М.: Наука, 1976. 304 с.

50. *Курант Р., Гильберт Д.* Методы математической физики. Т. 1. М.; Л.: Гостехиздат, 1951. 476 с.
51. *Лебедев В.И.* Функциональный анализ и вычислительная математика. М.: Физматлит, 2000. 296 с.
52. *Лионс Ж.-Л.* Некоторые методы решения нелинейных краевых задач. М.: Мир, 1972. 588 с.
53. *Локуциевский О.М., Гавриков М.Б.* Начала численного анализа. М.: ТОО «Янус», 1995. 581 с.
54. *Марчук Г.И.* Методы вычислительной математики. М.: Наука. Физматлит, 1989. 608 с.
55. *Марчук Г.И., Агошков В.И.* Введение в проекционно-сеточные методы. М.: Наука, 1981. 416 с.
56. *Михлин С.Г., Смолицкий Х.Л.* Приближенные методы решения дифференциальных и интегральных уравнений. М.: Наука. Физматлит, 1965. 384 с.
57. *Морозов В.А.* Регулярные методы решения некорректно поставленных задач. М.: Наука, 1987. 240 с.
58. *Никифоров А.Ф., Уваров В.Б.* Специальные функции математической физики. М.: Наука. Физматлит, 1984. 344 с.
59. *Никольский С.М.* Квадратурные формулы. М.: Наука, 1974. 223 с.
60. *Орtega Дж.* Введение в параллельные и векторные методы решения линейных систем. М.: Мир, 1991. 367 с.
61. *Орtega Дж., Рейнболдт В.* Итерационные методы решения нелинейных систем уравнений со многими неизвестными. М.: Мир, 1975. 560 с.
62. *Пирумов У.Г.* Численные методы. М.: Изд-во МАИ, 1998. 188 с.

63. *Писанецки С.* Технология разреженных матриц. М.: Мир, 1988. 411 с.
64. *Ректорис К.* Вариационные методы в математической физике и технике. М.: Мир, 1985. 590 с.
65. *Рябенький В.С.* Введение в вычислительную математику. М.: Наука. Физматлит, 1994. 336 с.
66. *Самарский А.А.* Введение в численные методы. М.: Наука. Физматлит, 1987. 288 с.
67. *Самарский А.А., Вабищевич П.Н.* Численные методы решения обратных задач математической физики. М.: Едиториал УРСС, 2004. 480 с.
68. *Самарский А.А., Гулин А.В.* Численные методы. М.: Наука. Физматлит, 1989. 416 с.
69. *Самарский А.А., Николаев Е.С.* Методы решения сеточных уравнений. М.: Наука, 1978. 592 с.
70. *Соболев С.Л.* Введение в теорию кубатурных формул. М.: Наука, 1975. 894 с.
71. *Соболь И.М.* Численные методы Монте-Карло. М.: Наука, 1973. 311 с.
72. *Стечкин С.Б., Субботин Ю.Н.* Сплайны в вычислительной математике. М.: Наука, 1976. 248 с.
73. *Съярле Ф.* Метод конечных элементов для эллиптических задач. М.: Мир, 1980. 512 с.
74. *Тихонов А.Н., Арсенин В.Я.* Методы решения некорректных задач. М.: Наука, 1979. 288 с.
75. *Тихонов А.Н., Гончарский А.В., Степанов В.В., Ягола А.Г.* Численные методы решения некорректных задач. М.: Наука, 1990. 232 с.

76. Тихонов А.Н., Самарский А.А. Уравнения математической физики. М.: Наука, 1972. 736 с.
77. Тьюарсон Р. Разреженные матрицы. М.: Мир, 1977. 189 с.
78. Уилкинсон Дж.Х. Алгебраическая проблема собственных значений. М.: Наука, 1970. 564 с.
79. Фаддеев Д.К., Фаддеева В.Н. Вычислительные методы линейной алгебры. М.; Л.: Физматгиз, 1963. 735 с.
80. Федоренко Р.П. Введение в вычислительную физику. М.: Изд-во МФТИ, 1994. 528 с.
81. Хайрер Э., Ваннер Г. Решение обыкновенных дифференциальных уравнений. Жесткие и дифференциально-алгебраические задачи. М.: Мир, 1999. 685 с.
82. Хайрер Э., Нёрсетт С., Ваннер Г. Решение обыкновенных дифференциальных уравнений. Нежесткие задачи. М.: Мир, 1990. 512 с.
83. Хейгеман Л., Янг Д. Прикладные итерационные методы. М.: Мир, 1986. 446 с.
84. Хорн Р., Джонсон Ч. Матричный анализ. М.: Мир, 1989. 655 с.
85. Эстербю О., Златев З. Прямые методы для разреженных матриц. М.: Мир, 1987. 118 с.

ПРЕДМЕТНЫЙ УКАЗАТЕЛЬ

Алгоритм:

насыщаемый 204
ненасыщаемый 204
неустойчивый 27
устойчивый 27

Аппроксимация разностного метода на точном решении 299
--- p -го порядка 299
Арифметика машинная 17
– конечная 17

Барьеры Бутчера 306

Вектор собственный 43

Выбор шага
автоматический 314
Выделение особенности:
аддитивное 279
мультипликативное 279

Граница оператора 42

Дефект сплайна 209

Дополнение ортогональное 34

Задача:

Коши 104, 295
– в интегральной форме 297
краевая 104, 339
Значение собственное 43, 44

Интерполянт 163

Интерполяция 163
кусочно-линейная 164
Лагранжа 172
обратная 183
полиномиальная 172
последовательная 212
с кратными узлами 184
тригонометрическая 205
Эрмита 184

Итерация:

внешняя 242
внутренняя 242
обратная 154

Константа Лебега 193

Корень квадратный
из оператора 42
Коэффициент Фурье 202

- Круги Гершгорина 58
- Мантисса** 17
- Матрица:
- вырожденная 57
 - Гильберта 81, 203
 - Грама 202
 - невырожденная 57
 - плохо обусловленная 74
 - предобуславливания 149
 - разреженная 157
- Метод:
- A*-устойчивый 332
 - $A(\alpha)$ -устойчивый 332
 - p*-го порядка точности 298
 - Адамса явный 318
 - бисекции 222
 - «вилки» 222
 - Галёркина 352
 - Гаусса 64
 - с выбором главного элемента 71
 - гибридный 249
 - Гира 318, 333
 - деления отрезка пополам 222
 - Зейделя 127
 - нелинейный 248
 - интерполяционный
 - решения нелинейных уравнений 230
 - итерационный 64, 119, 221
 - вариационного типа 138
 - двухслойный 119
 - двухшаговый 119, 146
 - нестационарный 120
 - неявный 120
 - одношаговый 119
 - стационарный 120
 - сходящийся 121, 221
 - трехслойный 119, 146
 - явный 120
 - касательных 234
 - квадратного корня 98
 - минимальных невязок 144
 - погрешностей 145
 - поправок 144
 - наименьших квадратов 171
 - Ньютона 234, 245
 - модифицированный 246
 - с параметром 246
 - парабол 240
 - Пикара 245
 - модифицированный 245
 - последовательного интегрирования 282
 - «предиктор — корректор» 302
 - прогонки 82
 - встречной 86
 - корректный 84, 96
 - левой 86

- матричной 96
- потоковой 87, 89
- правой 85
- пятидиагональной 94
- устойчивый 84, 96
- циклической 90, 93
- проекционно-сеточный 354
- проекционный 354
- простой итерации 122, 223
- прямой
 - решения СЛАУ 63
- разностный
 - A -устойчивый 332
 - $A(\alpha)$ -устойчивый 332
 - безусловно
 - устойчивый 327, 332
 - линейный
 - m -шаговый 315
 - нуль-устойчивый 325
 - условно устойчивый 327
- регуляризации
 - А.Н. Тихонова 157
 - СЛАУ 155
- релаксации 127, 245
 - нелинейный 226
- решения задачи Коши
 - p -го порядка
 - точности 298
 - – – неявный 316
 - – – полностью
 - неявный 316
- – – явный 316
- Ритца 352
- Ричардсона
 - с чебышёвскими параметрами 123
- Рунге — Кутты:
 - вложенный 313
 - двухстадийный 302
 - четвертого порядка 306
 - явный m -стадийный 303
- Рунге — Ромберга 288
- секущих 231
- скорейшего спуска 144
- сопряженных
 - градиентов 148
 - направлений 146
 - невязок 148
 - погрешностей 148
 - поправок 148
- степенной 154
- стрельбы 340
- типа простой итерации 223
- устойчивый безусловно 327
 - условно 327
- Холецкого 98
- хорд 231
- Эйлера 298
 - неявный 298
 - явный 298
- экстраполяции
 - Ричардсона 262

- Якоби 126
 – нелинейный 248
 ячеек 281
- Многообразие линейное 32
- Многочлен:
 интерполяционный 172
 – в форме Лагранжа 174
 – в форме Ньютона 180
 – Эрмита 184
- Фурье 202
- Невязка** 121, 140, 299
- Некорректность численного дифференцирования 290
- Неравенство
 Коши — Буняковского 34
- Норма 32
 вектора гильбертова 48
 – евклидова 48
 – естественная 34
 – кубическая 45
 – октаэдрическая 47
 – порожденная скалярным произведением 34
 – шаровая 48
 интерполяционного полинома 193
 матрицы максимальная 52
 – матричная 51
 – согласованная 50
 – спектральная 53
- Фробениуса 49
 оператора подчиненная 38
 Нормы эквивалентные 33
- Область:**
 значений оператора 37
 определения оператора 37
 устойчивости
 метода решения
 задачи Коши 330
- Ограничение функции
 на сетку 163
- Оператор:**
 кососимметричный 41
 линейный 37
 – ограниченный 38
 неотрицательный 41
 непрерывный 38
 нормальный 41
 обратный 39
 положительно
 определенный 41
 положительный 41
 самосопряженный 40
 сжимающий 40, 243
 сопряженный 40
- Операторы:**
 коммутирующие 39
 перестановочные 39
- Операция ограничения**
 функции на сетку 163

- Определитель
 Вандермонда 172
- Оценка погрешности:
 абсолютной вычисления
 функции линейная 30
 — — — предельная 30
 интерполяции наилучшая
 равномерная 192
- Параметр регуляризации** 156
- Погрешность:
 абсолютная 22
 алгоритма 27
 аппроксимации 299
 — правой части
 уравнения 299
 вычислений 16
 вычисления функции 29
 квадратурной формулы 256
 неустранимая 16
 относительная 22
 приближения 120, 221
 численного метода 16
 — — решения
 задачи Коши 298
- Подпространство
 линейное 32
- Полином:
 базисный 174
 — тригонометрический 205
 интерполяционный 172
- в форме Лагранжа 174
 — в форме Ньютона 180
 — Эрмита 184
 наименее уклоняющийся
 от нуля 191
 обобщенный 200
 Фурье 202
 Чебышёва 125, 191
- Поправка решения 140
- Последовательность:
 сходящаяся 32
 фундаментальная 33
- Правило Рунге:
 для квадратурных
 формул 262
- оценки погрешности метода
 решения ОДУ 311
- Предобуславливание 149
- Предобуславливатель 149
- Якоби 150
 блочный 150
 — Якоби 150
 диагональный 150
 неполный LU 151
- Признак Адамара 57, 85
- Принцип сжимающих
 отображений 40, 243
- Проблема собственных
 значений:
 ограниченная 153
 полная 153

- частичная 153
- Прогонка корректная 84
- матричная 97
- пятидиагональная 95
- устойчивая 84
- Производная разностная:
- вперед 287
 - вторая 290
 - левая 287
 - назад 287
 - правая 287
 - центральная 287
- Пространство линейное 31
- n -мерное 32
 - гильбертово 34
 - евклидово 34
 - нормированное 32
 - банахово 33
 - полное 33
 - Соболева 37
 - унитарное 34
 - энергетическое 42
- Процесс:
- итерационный
 - сходящийся 121
- Эйткена 290, 311
- Радиус оператора**
- спектральный 39
- Разложение матрицы
- неполное LU 151
- Разность разделенная 178
- Регуляризация
- дифференцирования 291
- Решения уравнения линейно независимые 104
- Сетка** 163
- прямоугольная 211
- Система:
- ОДУ жесткая 329
 - устойчивая
 - асимптотически 332
 - – безусловно 332
 - ортонормированная 34
 - полная 34
 - решений линейного разностного уравнения
 - функций чебышёвская 173
 - устойчивая
 - по правой части 74
 - скалярное произведение 33
 - Скорость сходимости 121
 - Спектр оператора 43
- Сплайн:
- B*-сплайн 211
 - естественный 208
 - интерполяционный
 - кубический 208
 - степени m 209
 - чертежный 208

- Сумма квадратурная 256
- Схема:
- с экспоненциальной подгонкой 347
 - симметричные решения задачи Коши 300
- Сходимость метода:
- линейная со скоростью геометрической прогрессии 225
 - решения задачи Коши 298
 - с p -м порядком 236
 - стационарного 242
 - типа простой итерации 223
- Таблицы Бутчера** 307
- Теорема:
- Вейерштрасса 172
 - Гершгорина первая 58
 - вторая 58
 - Марцинкевича 195
 - Фабера 195
- Точка оператора
- неподвижная 40
- Узел сетки** 163
- Уравнение:
- Далквиста тестовое 330
 - линейное разностное 101
 - однородное 101
 - неоднородное 101
- с постоянными коэффициентами 101
- характеристическое m -шагового разностного метода 325
- линейного разностного однородного уравнения 105
- Условие диагонального преобладания 84
- корней 325
- Условия согласования 304
- точности квадратурной формулы 268
- Уточнение итерационное 151
- Форма записи каноническая** двухслойного итерационного метода 119
- Формула:
- Гаусса 271
 - дифференцирования назад 318
 - квадратурная 256
 - l -го порядка точности 258
 - Гаусса 267, 271
 - Ньютона — Котеса 266
 - Симпсона 260
 - Филона 276
 - интерполяционного типа 257, 263

- наивысшей алгебраической степени 271
- прямоугольников левых 258
- – правых 258
- – центральных 257
- трапеций 259
- Формулы обратного хода 84
- прямого хода 83
- Функция:
 - базисная 166
 - сеточная 163
 - формы 166
- Цифра** числа значащая 23
- – – верная 23
- Число:**
 - жесткости 329
 - на временном интервале 329
 - обусловленности 74
 - с плавающей запятой 17
- наивысшей алгебраической степени 271
- – – нормализованное 18
- сингулярное 52
- собственное 43
- Член остаточный
- интерполяционного полинома 174
- Экстраполяция** 164
 - двумерная 215
 - по Ричардсону 311
- Элемент:**
 - конечный одномерный
 - линейный 166
 - двумерный 215
 - наилучшего приближения 201
 - сетки 163
 - собственный 43, 45
- Элементы:**
 - взаимно ортогональные 34
 - линейно зависимые 32
 - независимые 32
- Ядро** оператора 39

ОГЛАВЛЕНИЕ

Предисловие	3
Основные обозначения	7
Введение	11
1. Предварительные сведения	15
1.1. Погрешности при вычислениях	15
1.1.1. Причины появления погрешностей	15
1.1.2. Хранение чисел на ЭВМ и погрешности округления	17
1.1.3. Погрешности арифметических операций.....	24
1.1.4. Погрешность алгоритма	27
1.2. Элементы функционального анализа и линейной алгебры.....	31
1.2.1. Линейные пространства	31
1.2.2. Примеры нормированных линейных пространств	35
1.2.3. Операторы в нормированных пространствах....	37
1.2.4. Операторы в гильбертовых пространствах.....	40
1.2.5. Операторы в конечномерных пространствах	43
1.2.6. Нормы векторов и матриц.....	45
1.2.7. Геометрическая интерпретация понятия линейного оператора.....	55
1.2.8. Признак Адамара и теоремы Гершгорина.....	56
Вопросы и задания.....	61
Библиографические комментарии	62
2. Прямые методы решения систем линейных алгебраических уравнений.....	63

2.1. Постановка задачи.....	63
2.2. Метод Гаусса	64
2.2.1. Схема метода Гаусса	64
2.2.2. Расчетные формулы и количество действий метода Гаусса	66
2.2.3. Связь метода Гаусса с разложением матрицы на множители	69
2.2.4. Выбор главного элемента.....	70
2.3. Обусловленность систем линейных алгебраических уравнений	74
2.4. Метод прогонки	82
2.4.1. Метод правой прогонки	82
2.4.2. Методы левой и встречных прогонок	86
2.4.3. Метод потоковой прогонки	87
2.4.4. Метод циклической прогонки	90
2.4.5. Метод пятидиагональной прогонки.....	94
2.4.6. Метод матричной прогонки.....	96
2.5. Метод квадратного корня.....	98
2.6. Решение линейных разностных уравнений	101
2.6.1. Линейные разностные уравнения.....	101
2.6.2. Линейные разностные уравнения с постоянными коэффициентами	105
Вопросы и задания.....	115
Библиографические комментарии	116
3. Итерационные методы решения систем линейных алгебраических уравнений	118
3.1. Классические одношаговые итерационные методы ...	118
3.1.1. Каноническая форма одношаговых итерационных методов	118
3.1.2. Одношаговые итерационные методы.....	122

3.1.3. Геометрическая интерпретация одношаговых стационарных итерационных методов	129
3.1.4. Условия сходимости стационарных итерационных методов	133
3.2. Итерационные методы вариационного типа	138
3.2.1. Вариационный подход к построению итерационных методов	138
3.2.2. Расчетные формулы методов вариационного типа	140
3.2.3. Оценка скорости сходимости	141
3.2.4. Частные случаи методов вариационного типа ..	144
3.3. Методы сопряженных направлений	146
3.4. Предобуславливание	149
3.5. Итерационное уточнение решения	151
3.6. Решение проблемы собственных значений	153
3.7. Регуляризация плохо обусловленных систем линейных алгебраических уравнений	155
3.8. Хранение больших разреженных матриц	157
Вопросы и задания	158
Библиографические комментарии	160
4. Методы интерполяирования функций	162
4.1. Постановка задачи и простейшие методы интерполяирования функций	162
4.1.1. Основные определения	162
4.1.2. Кусочно-линейная интерполяция	164
4.1.3. Многовариантность интерполяирования	167
4.2. Полиномиальная интерполяция	172
4.2.1. Обоснование полиномиальной интерполяции ..	172
4.2.2. Интерполяционный полином в форме Лагранжа	174

4.2.3. Интерполяционный полином в форме Ньютона	178
4.2.4. Интерполяционный полином Эрмита	183
4.3. Сходимость и устойчивость полиномиальной интерполяции	190
4.3.1. Оптимизация узлов сетки	190
4.3.2. Устойчивость интерполяционного полинома относительно погрешностей функции	193
4.3.3. Устойчивость интерполяционного полинома относительно априорной информации	194
4.3.4. Наилучшие приближения в гильбертовом пространстве	200
4.3.5. Насыщаемость алгоритма интерполяции. Тригонометрическая интерполяция	204
4.4. Сплайн-интерполяция	206
4.5. Двумерная интерполяция.....	211
Вопросы и задания.....	216
Библиографические комментарии	217
5. Решение нелинейных уравнений	219
5.1. Решение скалярных уравнений	219
5.1.1. Постановка задачи и основные процедуры решения.....	219
5.1.2. Метод «вилки», или деления отрезка пополам ..	222
5.1.3. Итерационные методы типа простой итерации ..	223
5.1.4. Интерполяционные методы	230
5.2. Решение систем нелинейных уравнений	241
5.2.1. Постановка задачи и основные понятия	241
5.2.2. Сходимость стационарного метода	242
5.2.3. Примеры итерационных методов	245
Вопросы и задания.....	252
Библиографические комментарии	253

6. Методы численного интегрирования и дифференцирования.....	255
6.1. Простейшие квадратурные формулы.....	255
6.1.1. Постановка задачи и основные определения.....	255
6.1.2. Формула прямоугольников	257
6.1.3. Формула трапеций	259
6.1.4. Формула Симпсона.....	260
6.2. Квадратурные формулы интерполяционного типа.....	262
6.3. Квадратурные формулы Гаусса	267
6.4. Интегрирование быстроосцилирующих функций	275
6.5. Вычисление несобственных интегралов I и II рода.....	276
6.6. Вычисление кратных интегралов	281
6.7. Численное дифференцирование.....	283
Вопросы и задания.....	292
Библиографические комментарии	293
7. Численное решение задачи Коши для обыкновенных дифференциальных уравнений.....	295
7.1. Постановка задачи.....	295
7.2. Простейшие методы численного решения задачи Коши	297
7.2.1. Методы Эйлера.....	297
7.2.2. Симметричная схема	300
7.2.3. Метод Рунге — Кутты второго порядка	301
7.3. Методы Рунге — Кутты	302
7.3.1. Явные методы Рунге — Кутты.....	302
7.3.2. Доказательство сходимости методов Рунге — Кутты.....	308
7.3.3. Управление длиной шага	311
7.4. Многошаговые разностные методы	315
7.4.1. Определение линейных многошаговых методов	315

7.4.2. Погрешность аппроксимации многошаговых методов	318
7.4.3. Примеры методов Адамса и Гира.....	320
7.4.4. Устойчивость и сходимость разностных методов	324
7.5. Методы решения жестких систем	327
7.5.1. Условно устойчивые и безусловно устойчивые разностные методы.....	327
7.5.2. Понятие жесткой системы ОДУ	328
7.5.3. Решение жестких систем.....	329
Вопросы и задания.....	336
Библиографические комментарии	338
8. Решение краевых задач для систем обыкновенных дифференциальных уравнений.....	339
8.1. Постановка задачи. Метод стрельбы	339
8.2. Конечно-разностные методы	345
8.2.1. Линейная краевая задача второго порядка	345
8.2.2. Нелинейные задачи	348
8.3. Методы Ритца и Галёркина.....	350
Вопросы и задания.....	354
Библиографические комментарии	355
Литература	356
Предметный указатель	363