КАЗАНСКИЙ ФЕДЕРАЛЬНЫЙ УНИВЕРСИТЕТ

Р.З. Даутов, М.Р. Тимербаев

Численные методы.
Задачи линейной алгебры и дифференциальных уравнений

Учебное пособие

Казань — 2021

УДК 519.3

Рекомендовано
учебно–методической комиссией института ВМ и ИТ
Протокол 8 от 19 марта 2021 г.

Научный редактор —
д.ф.-м.н., профессор О.А. Задворнов

Рецензенты:
д.ф.-м.н., профессор М.Ф. Павлова,
к.ф.-м.н., доцент Л.Л. Глазырина

Р.З. Даутов, М.Р. Тимербаев
Численные методы. Решение задач линейной алгебры и дифференциальных уравнений: учебное пособие. — Казань: К(П)ФУ, 2021. — 168 с.

Учебное пособие представляет собой вторую часть курса лекций по дисциплине “Численные методы“, читаемого авторами студентам Института вычислительной математики и информационных технологий Казанского (Приволжского) федерального университета. В нем изложены численные методы решения основных задач линейной алгебры, а также методы решения дифференциальных уравнений, как обыкновенных, так и в частных производных. Приведены вопросы для самоконтроля и задачи по каждой теме. Пособие предназначено для студентов-бакалавров, изучающих численные методы.

УДК 519.3

Оглавление

ГЛАВА 1. **Прямые методы решения систем линейных уравнений** . 5
§ 1. Трудоемкость базовых операций линейной алгебры. 9
§ 2. Простые системы уравнений 12
§ 3. Метод исключения Гаусса 16
§ 4. Компактные схемы метода Гаусса 27
§ 5. Элементарные ортогональные матрицы 32
§ 6. QR разложение матриц 36
§ 7. Вычисление определителей и обратной матрицы 38
§ 8. Решение разреженных систем уравнений 40
§ 9. О технологии разреженных матриц 44
§ 10. Метод прогонки 47
§ 11. Нормы векторов и матриц 49
§ 12. Устойчивость решений СЛАУ 57

ГЛАВА 2. **Итерационные методы решения систем уравнений** . . . 62
§ 1. Простейшие итерационные методы. 63
§ 2. Элементы общей теории итерационных методов 66
§ 3. Достаточное условие сходимости при $A = A^T > 0$ 71
§ 4. Оптимальный выбор итерационного параметра 73
§ 5. Критерии остановки итераций и выбор матрицы B 76
§ 6. Итерационные методы вариационного типа 79
§ 7. Метод сопряженных градиентов. 85

ГЛАВА 3. **Алгебраическая проблема собственных значений** 89
§ 1. Степенной метод и метод обратных итераций 92
§ 2. Метод вращений (Якоби) 96
§ 3. Понятие о QL-методе 99

ГЛАВА 4. **Решение нелинейных уравнений** 103
§ 1. Вычисление нулей функции одной переменной 103
§ 2. Методы решения систем нелинейных уравнений 109

ГЛАВА 5. **Методы решения задачи Коши для систем ОДУ** 113
§ 1. Семейство методов Рунге – Кутты. 120
§ 2. Двухстадийные методы Рунге — Кутты 122
§ 3. Оценка точности методов Рунге — Кутты 124
§ 4. Многошаговые методы 130
§ 5. Явные методы Адамса 132
§ 6. Невные методы Адамса 135
§ 7. Устойчивость явных и неявных методов 136

ГЛАВА 6. **Методы решения одномерных краевых задач** 140
§ 1. Метод коллокации 140
§ 2. Методы конечных разностей 142
§ 3. Метод Галеркина 146
§ 4. Метод конечных элементов 150

ГЛАВА 7. **Решение уравнений в частных производных** 154
§ 1. Разностные методы для уравнения теплопроводности 154
§ 2. Разностная схема для уравнения Пуассона 160

Литература 168

ГЛАВА 1

Прямые методы решения систем линейных уравнений

Многие задачи практики приводят к необходимости решать системы линейных алгебраических уравнений (СЛАУ). При конструировании инженерных сооружений, приборов, обработке результатов измерений, решении задач планирования производственного процесса и многих других задач техники, экономики, научного эксперимента приходится решать СЛАУ.

Совокупность линейных соотношений

$$\begin{cases} a_{11}x_1 + a_{12}x_2 + \ldots + a_{1n}x_n = b_1, \\ a_{21}x_1 + a_{22}x_2 + \ldots + a_{2n}x_n = b_2, \\ \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots \\ a_{n1}x_1 + a_{n2}x_2 + \ldots + a_{nn}x_n = b_n, \end{cases}$$

как мы знаем, называется системой линейных алгебраических уравнений. Компактно эти уравнения записываются в виде[1)]

$$\sum_{j=1}^{n} a_{ij}\, x_j = b_i, \quad i = 1 : n,$$

а также в матричном виде

$$Ax = b.$$

Здесь

$$A = \begin{pmatrix} a_{11} & a_{12} & \ldots & a_{1n} \\ a_{21} & a_{22} & \ldots & a_{2n} \\ \vdots & \vdots & & \vdots \\ a_{n1} & a_{n2} & \ldots & a_{nn} \end{pmatrix}, \quad x = \begin{pmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \end{pmatrix}, \quad b = \begin{pmatrix} b_1 \\ b_2 \\ \vdots \\ b_n \end{pmatrix}.$$

[1)] Запись $i = 1 : n$ означает, что $i = 1, 2, \ldots, n$.

Матрица коэффициентов A и вектор правой части b системы уравнений (столбец свободных членов) являются данными, а вектор неизвестных x требуется определить.

Исследование ряда научно-технических, экономических и прочих проблем приводит к математическим моделям непосредственно в форме систем линейных алгебраических уравнений. Однако гораздо чаще они появляются в процессе математического моделирования как промежуточный этап при решении более сложной задачи, например, после дискретизации (и, если необходимо, линеаризации) интегральных, дифференциальных, интегро-дифференциальных уравнений или систем уравнений такого сорта. В силу этого задачи линейной алгебры являются наиболее часто решаемыми математическими задачами в процессе математического моделирования.

1. Разрешимость СЛАУ. Итак, рассмотрим СЛАУ

$$\sum_{j=1}^{n} a_{ij}\, x_j = b_i, \quad i = 1 : n,$$

которая в матричном виде записывается как

$$Ax = b. \tag{1}$$

Далее мы будем рассматривать только случай, когда данные задачи являются вещественными (т.е. $A \in \mathbb{R}^{n\times n}$, $b \in \mathbb{R}^n$), а сама задача однозначно разрешима.

В следующей теореме собраны основные критерии однозначной разрешимости СЛАУ.

Лемма 1. Следующие утверждения эквивалентны:
1) решение системы $Ax = b$ существует и единственно при любом b;
2) определитель A отличен от нуля (т.е. $\det(A) \neq 0$).[1)]
3) однородная система $Ax = 0$ имеет лишь решение $x = 0$;
4) система $Ax = b$ имеет единственное решение;
5) матрица A обратима (т.е. существует матрица A^{-1});
6) $\operatorname{rank}(A) = n$.

[1)] Для определителя матрицы A будет использоваться также обозначение $|A|$.

2. О методе Крамера. Решение системы (1) может быть выписано по формулам Крамера

$$x_i = \frac{\Delta_i}{\Delta}, \quad i = 1 : n,$$

где $\Delta = \det(A)$, а Δ_i — определитель, получающийся из определителя матрицы A заменой i-того столбца столбцом свободных членов b.

Казалось бы, формулы Крамера полностью решают задачу построения решения системы линейных уравнений. Однако на практике они не используются. Это объясняется следующим. Напомним, что

$$\det(A) = \sum_{(\alpha_1,\alpha_2,\ldots,\alpha_n)} \pm\, a_{1\alpha_1} a_{2\alpha_2} \cdots a_{n\alpha_n},$$

где $(\alpha_1, \alpha_2, \ldots, \alpha_n)$ — перестановка чисел $(1\ 2\ \ldots\ n)$. Число слагаемых в сумме равно $n!$, поэтому непосредственное вычисление определителя требует более $n!$ арифметических операций (короче — flop: floating point operation), что уже при $n = 30$ недоступно даже для самых мощных ЭВМ. Поэтому для решения систем уравнений применяют другие, более экономичные методы.

3. Типы матриц и методов. Для большинства вычислительных задач, встречающихся на практике, характерным является *большой порядок* n матрицы A, а также *серийность* задачи: требуется решить не одну, а целую серию СЛАУ с одной и той же или близкими матрицами и с разными правыми частями. В связи с этим, там где это возможно, мы будем указывать оценки трудоемкости описываемых методов. Они имеют важное значение для сравнительного анализа численных методов решения задач линейной алгебры.

Вопрос: каких значений может достигать величина n? С системами уравнений какой размерности приходится иметь дело в приложениях на сегодняшний день?

Ответ: величина n может достигать значений $10^6 \sim 10^7$ и есть большая необходимость в решении систем большего порядка.

Системы такой размерности трудно себе представить! Еще труднее себе представить как такие системы решаются. В самом деле, если хранить матрицу A в памяти ЭВМ, то на это потребуется $Q = 8n^2$ байт памяти, если для хранения одного числа отводится 8 байт (как

для чисел типа *double*). При $n = 10^6$ получаем $Q = 8 \cdot 10^{12}$ байт или $Q = 8$ Терабайт. Можно предположить, что для решения СЛАУ такой размерности нужны суперкомпьютеры. Однако, это не так по нескольким причинам. Укажем их.

i) Заполненность матриц. В приложениях приходится иметь дело с двумя типами матриц: с *плотными* и *разреженными* матрицами.

Матрицы, наличием нулевых элементов в которых можно пренебречь, называют *плотными матрицами*. Они хранятся в ЭВМ в виде двумерного массива и обращение к элементу a_{ij} этой матрицы требует небольших накладных расходов.

Матрицы, содержащие относительно небольшое число ненулевых элементов называют *разреженными матрицами*. Хорошим примером такой матрицы является матрица достаточно большого размера n (например, $n \approx 10^6$), на каждой строке которой имеется лишь небольшое число m ненулевых элементов (например, $m \approx 10 \sim 100$). Такие матрицы хранятся в памяти ЭВМ в специальном формате, причем хранятся только ненулевые элементы. Например, для их хранения при $n = 10^6$, $m \approx 10$ требуется ≈ 100 Мегабайт памяти. Обращение к произвольному элементу a_{ij} такой матрицы требует большèх накладных расходов, но, в зависимости от формата хранения, требует небольших расходов для доступа ко всем ненулевым элементам строки или столбца.

В приложениях плотных матриц порядки $n \approx 10^4$ считаются большими, а $n \approx 10^5$ — сверхбольшими. Для разреженных матриц сверхбольшими считаются порядки $n \approx 10^6 \sim 10^7$.[1)]

ii) Типы численных методов. Методы решения СЛАУ делятся на прямые и итерационные.

Метод называется *прямым*, если для нахождения решения требуется конечное число арифметических операций (+,-,*,/) и извлечений квадратного корня. Например, метод Крамера является прямым методом :-)). Прямые методы требуют хранения матрицы A, а также некоторый объем накладной памяти.

Итерационные методы позволяют за конечное число операций отыскать лишь приближенное решение (с заданной точностью). Они

[1)]Надо понимать, что эти градации являются относительными. Все зависит от конкретной задачи, длины серии, доступного компьютера и программного обеспечения и т.д.

реализуются, чаше всего, как одношаговые или двухшаговые рекуррентные формулы и генерируют последовательность векторов-приближений, сходящуюся к решению. Итерационные методы не требуют обязательного хранения матрицы A; требуется уметь вычислять лишь произведение A на заданный вектор (не редки случаи, когда это можно сделать без хранения A).

iii) Точность решения. Необходимо отметить, что прямые методы только теоретически позволяют найти точное решение задачи, поскольку числа в ЭВМ представляются приближенно (с конечным числом разрядов). Для чисел типа double относительная точность представления равна $2.2 \cdot 10^{-16}$ (т.е. числа типа double имеют около 16 верных значащих десятичных цифр).

Кроме того, для плотных матриц прямые методы требуют порядка $O(n^3)$ flop, каждая из которых также приводит к появлению погрешности в вычислениях. Из-за большого числа операций это приводит к некоторому, а иногда и заметному, накоплению погрешностей округления. Таким образом, в практических вычислениях прямые методы также позволяют найти решение СЛАУ лишь приближенно, хотя, как правило, с высокой точностью (это зависит от свойств матрицы). Отметим, что в практических ситуациях не всегда нужна высокая точность решения.

Итерационные методы оказываются выгодными, если: а) нужна невысокая точность решения; б) при решении дольших и сверхбольших систем; с) при решении СЛАУ со специальными матрицами.

Из-за ограниченности времени и трудности проблемы, вопросов накопления погрешности в вычислениях далее мы касаться не будем. В определенных ситуациях ограничимся лишь замечаниями.

Будем считать далее, что все вычисления осуществляются в точной арифметике.

§ 1. Трудоемкость базовых операций линейной алгебры.

Рассмотрим предварительно трудоемкость некоторых операций.

1. Вычисление суммы векторов. Пусть требуется вычислить сумму $z = x + y$ двух векторов x и y размера n. По определению

$$z_i = x_i + y_i, \quad i = 1 : n.$$

Ясно, что трудоемкость метода составляет n flop.

2. Вычисление скалярного произведения. Трудоемкость вычисления скалярного произведения $(x, y) = \sum_{i=1}^{n} x_i\, y_i$ векторов x и y составляет $2n$ flop (n умножений и n сложений).

3. Вычисление произведения матрицы и вектора. Пусть заданы матрица A размера n и вектор x. Рассмотрим задачу вычисления вектора $b = Ax$. По определению

$$\begin{aligned} b_1 &= a_{11}x_1 + a_{12}x_2 + \ \ldots\ + a_{1n}x_n, \\ b_2 &= a_{21}x_1 + a_{22}x_2 + \ \ldots\ + a_{2n}x_n, \\ &\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots \\ b_n &= a_{n1}x_1 + a_{n2}x_2 + \ \ldots\ + a_{nn}x_n, \end{aligned}$$

или короче,

$$b_i = \sum_{j=1}^{n} a_{ij}\, x_j, \quad i = 1 : n. \tag{2}$$

Будем говорить, что формула (2) определяет *метод* умножения матрицы A на заданный вектор. Заметим, что он ориентирован на работу со строками матрицы и определяет b_i как скалярное произведение i-той строки A на вектор-строку x.

Непосредственная реализация формул (2) в MATLAB приводит к следующей функции:

```
function b = Axrow(A,x)
n = numel(x);
b = zeros(size(x));
for i=1:n
  for j=1:n
    b(i) = b(i) + A(i,j)*x(j);
  end
end
```

В этой функции компоненты вектора b вычисляются последовательно друг за другом накоплением. Здесь и далее цикл по i означает цикл по строкам, а цикл по j — цикл по столбцам матрицы. Нетрудно видеть, что трудоемкость этой функции равна $2n^2$ flop.

Алгоритм вычисления, реализованный в функции Axrow, называют строчно-ориентированным: в нем цикл по i предшествует циклу по j и в нем обрабатываются в цикле по j строки матрицы. Поменяв порядок циклов придем к другой реализации формул (2) (столбцово-ориентированной). В нем цикл по j предшествует циклу по i:

```
function b=Axcol(A,x)
n = numel(x);
b = zeros(size(x));
for j=1:n
  for i=1:n
    b(i) = b(i) + A(i,j)*x(j);
  end
end
```

В функции Axcol накоплением вычисляются вклады произведения Ax сразу во все компоненты вектора b и ее трудоемкость также равна $2n^2$ flop. В этой функции непосредственно реализован способ вычисления b, основанный на эквивалентной (2) формуле и ориентированной на столбцы матрицы. Он имеет следующий вид:

$$b = \begin{bmatrix} b_1 \\ b_2 \\ \vdots \\ b_m \end{bmatrix} = \begin{bmatrix} a_{11} \\ a_{21} \\ \vdots \\ a_{n1} \end{bmatrix} x_1 + \begin{bmatrix} a_{12} \\ a_{22} \\ \vdots \\ a_{n2} \end{bmatrix} x_2 + \begin{bmatrix} a_{1n} \\ a_{2n} \\ \vdots \\ a_{nn} \end{bmatrix} x_n.$$

Вместо языка MATLAB можно взять другой язык программирования (например, СИ, Паскаль, Фортран, ...) и написать аналоги функций Axrow и Axcol на этом языке. Практический интерес представляет ответ на следующий вопрос: *какая из полученных функций будет быстрее*, т. е. будет требовать меньшего времени для выполнения? На первый взгляд время работы функций не должно различаться. Однако это не так. *Ответ на этот важный с практической точки зрения вопрос зависит от языка программирования* и связан, главным образом, со способом хранения матриц (способом адресации элементов матриц). Из-за наличия в современных компьютерах многоуровнего кэша последовательное извлечение из оперативной памяти и сохранение чисел, расположенных в соседних ячейках памяти, производится намного быстрее, чем последовательное выполнение тех же операций с элементами, расположенными в памяти далеко друг от друга. В связи с этим, если элементы матрицы в памяти ЭВМ хранятся по строкам (как, например, в C, Паскаль, Python), то быстрее будет выполняться строчно-ориентированная функция Axrow. И наоборот, если элементы матрицы в памяти ЭВМ хранятся по столбцам (Fortran, MATLAB, OpenGL), то быстрее будет выполняться столбцово-ориентированная функция Axcol.

Будем говорить, что функции Axrow и Axcol реализуют *алгоритм* умножения матрицы A на заданный вектор. Эти функции демонстрируют разницу между методом и алгоритмом. В дальнейшем мы ограничимся указанием лишь метода решения задачи.

4. Вычисление произведения двух матриц. Пусть требуется вычислить произведения $C = AB$ двух заданных матриц размера n. По определению j-й столбец C есть произведение матрицы A и j-го столбца B. Так, если b_j есть j-тый столбец B, то $C = AB = A[b_1, b_2, \ldots, b_n] = [Ab_1, Ab_2, \ldots, Ab_n]$, или

$$c_{ij} = \sum_{k=1}^{n} a_{ik} b_{kj}, \quad i, j = 1 : n.$$

Вычисление c_{ij} (скалярного произведения i-той строки A на j-тый столбец B) требует $2n$ flop, и нужно вычислить n^2 таких элементов. Поэтому трудоемкость вычисления C равна $2n^3$ flop.

§ 2. Простые системы уравнений

Приведем примеры систем уравнений, которые легко решаются.

1. Системы с диагональной матрицей. Пусть D есть диагональная матрица с ненулевыми элементами d_i на диагонали, т. е. $D = \operatorname{diag}(d_1, d_2, \ldots, d_n)$. Тогда система уравнений $Dx = b$ элементарно решается за n flop, и компоненты вектора x находятся по формулам $x_i = b_i/d_i$, $i = 1 : n$.

2. Системы с треугольной матрицей. Матрица A называется *нижней треугольной* (также левой треугольной), если $a_{ij} = 0$ при всех $j > i$. Аналогично, матрица A называется *верхней треугольной* (также правой треугольной), если $a_{ij} = 0$ при всех $i > j$. Как правило, нижние треугольные матрицы обозначаются буквой L (Lower, Left), а верхние треугольные — буквой U (Upper) или R (Right). Таким образом,

$$L = \begin{bmatrix} l_{11} & 0 & \ldots & 0 \\ l_{21} & l_{22} & \ldots & 0 \\ \vdots & \vdots & & \vdots \\ l_{n1} & l_{n2} & \ldots & l_{nn} \end{bmatrix}, \quad U = \begin{bmatrix} u_{11} & u_{12} & \ldots & u_{1n} \\ 0 & u_{22} & \ldots & u_{2n} \\ \vdots & \vdots & & \vdots \\ 0 & 0 & \ldots & u_{nn} \end{bmatrix}.$$

Матрица называется треугольной, если она является либо нижней треугольной, либо верхней треугольной.

Поскольку определитель L равен $\det(L) = l_{11}l_{22}\cdots l_{nn}$, и аналогично $\det(U) = u_{11}u_{22}\cdots u_{nn}$, то треугольные матрицы невырождены тогда и только тогда, когда все их диагональные элементы отличны от нуля.

Система уравнений $Lx = b$ в индексной форме имеет вид

$$\begin{aligned} &l_{11}x_1 && = b_1,\\ &l_{21}x_1 + l_{22}x_2 && = b_2,\\ &\ldots\ldots\ldots\ldots && \ldots\\ &l_{n1}x_1 + l_{n2}x_2 + \ldots + l_{nn}x_n && = b_n. \end{aligned}$$

Решение этой системы находится последовательно: из первого уравнения определяется $x_1 = b_1/l_{11}$, из второго $x_2 = (b_2 - l_{21}x_1)/l_{22}$ и т. д. Таким образом,

$$x_i = \Big(b_i - \sum_{j=1}^{i-1} l_{ij}\, x_j\Big)/l_{ii}, \quad i = 1:n. \tag{3}$$

При $i = 1$ в (3) возникает сумма $\sum\limits_{j=1}^{0} \big(\ldots\big)$. Подобные суммы здесь и далее считаются равными нулю.

Метод (3) решения системы $Lx = b$ называется *прямой подстановкой*. Определим трудоемкость этого метода: при фиксированном i требуется $2(i-1)$ flop для вычисления суммы и дополнительно 2 flop для вычисления x_i. Общее число операций равно

$$Q = 1 + 2\sum_{i=2}^{n} i = n^2 + n - 1 = n^2 + O(n)\ \text{flop}\,,$$

т.к. $1 + 2 + \ldots + m = m(m+1)/2$. Аналогично решается система $Ux = b$. Отличие в том, что сначала определяется $x_n = b_n/u_{nn}$, затем $x_{n-1} = (b_{n-1} - u_{n-1,n}x_n)/u_{n-1,n-1}$ и т. д. Таким образом,

$$x_i = \Big(b_i - \sum_{j=i+1}^{n} u_{ij}\, x_j\Big)/u_{ii}, \quad i = n, n-1, \ldots, 1. \tag{4}$$

Метод (4) решения системы $Ux = b$ называется *обратной подстановкой*. Его трудоемкость также равна $n^2 + O(n)$ flop.

Обратим внимание, что суммарная трудоемкость прямого и обратного хода, т. е. трудоемкость последовательного решения двух треугольных систем $Ly = b$ и $Ux = y$ равна $2n^2 + O(n)$ flop, и при больших значениях n примерно равна трудоемкости вычисления $b = Ax$ при заданном x.

Отметим также замкнутость множества $\mathcal{L}$ всех нижних треугольных матриц (множества $\mathcal{U}$ — всех верхних треугольных матриц) относительно операций сложения и умножения. В самом деле, пусть $L, L_1, L_2 \in \mathcal{L}$. Тогда $L_1 + L_2 \in \mathcal{L}$ (что очевидно), $L_1 L_2 \in \mathcal{L}$ (непосредственно проверяется) и, если L — обратим, то $L^{-1} \in \mathcal{L}$ (см. ниже задачу 6). По определению нулевая матрица и единичная матрица являются элементами как $\mathcal{L}$, так и $\mathcal{U}$. Кроме того, $L_1 + L_2 = L_2 + L_1$, но, вообще говоря, $L_1 L_2 \neq L_2 L_1$. Квадратная матрица

$$L_k = \begin{bmatrix} 1 & \cdots & 0 & 0 & \cdots & 0 \\ \cdots & \cdots & \cdots & \cdots & \cdots & \cdots \\ 0 & \cdots & l_{k,k} & 0 & \cdots & 0 \\ 0 & \cdots & l_{k+1,k} & 1 & \cdots & 0 \\ \cdots & \cdots & \cdots & \cdots & \cdots & \cdots \\ 0 & \cdots & l_{n,k} & 0 & \cdots & 1 \end{bmatrix} \tag{5}$$

называется *элементарной нижней треугольной*; она отличается от единичной матрицы лишь элементами k-го столбца. Важное свойство этой матрицы отмечено далее в задаче 5.

3. Системы с ортогональной матрицей. Вещественная матрица Q называется ортогональной, если $Q^T Q = QQ^T = I$, где T означает знак транспонирования, I — единичная матрица. Равенство $Q^T Q = E$ ($QQ^T = E$) согласно правилу умножения матриц означает, что столбцы (строки) Q образуют ортонормированную систему из n векторов. По определению ортогональной матрицы $Q^{-1} = Q^T$. [1)]

Пусть требуется решить систему $Qx = b$. Умножая обе части этого равенства на Q^T, получим $x = Q^T b$. Трудоемкость такого метода решения есть $2n^2$ flop, если Q есть плотная матрица.

Простейшим примером ортогональной матрицы является элементарная матрица перестановок (транспозиция). Матрица P_{ik} называется *элементарной матрицей перестановок*, если она получена из

[1)] Напомним, что B называется обратной к A, если $AB = BA$. Она обозначается через A^{-1}.

единичной матрицы перестановкой строк с номерами i и k. Например, матрицами перестановок третьего порядка являются матрицы

$$P_{12} = \begin{bmatrix} 0 & 1 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 1 \end{bmatrix}, \quad P_{13} = \begin{bmatrix} 0 & 0 & 1 \\ 0 & 1 & 0 \\ 1 & 0 & 0 \end{bmatrix}, \quad P_{23} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 0 & 1 \\ 0 & 1 & 0 \end{bmatrix}.$$

Далее мы встретимся еще двумя типами ортогональных матриц Q, таких, что $Q = Q^T = Q^{-1}$. Это матрицы отражения и вращения. Матрицы вращения, как и элементарная матрица перестановок являются разреженными (на каждой строке не более двух ненулевых элементов).

Имеется большое количество прямых методов решения СЛАУ. Далее мы ограничимся рассмотрением лишь двух семейств методов, основанных на идее треугольной и ортогональной факторизации матриц: вариантов метода Гаусса и QR разложения.

Задания для самостоятельной работы

ВОПРОСЫ ДЛЯ САМОКОНТРОЛЯ

1. Что в линейной алгебре понимают под: a) вектором-столбцом? b) вектором-строкой? c) матрицей?

2. Что понимается под решением СЛАУ?

3. Приведите критерии разрешимости СЛАУ?

4. Укажите формулы Крамера решения СЛАУ.

5. Какие матрицы называются: а) плотными? б) разреженными?

6. Укажите формулы и трудоемкость следующих операций: а) суммы векторов; б) произведения матрицы и вектора; с) произведения двух матриц.

7. Какая матрица называется: а) диагональной; б) нижней треугольной; с) верхней треугольной; д) ортогональной;

8. Укажите формулы решения системы: а) с нижней треугольной матрицей; б) с верхней треугольной матрицей; с) с ортогональной матрицей.

9. Укажите трудоемкость решения системы: а) с нижней треугольной матрицей; б) с верхней треугольной матрицей; с) с ортогональной матрицей.

10. В чем различие между методом и алгоритмом? Приведите примеры.

11. Дайте определение элементарной нижней треугольной матрицы.

12. Дайте определение матрицы перестановок.

13. Какие свойства матрицы перестановок можно отметить?

ЗАДАЧИ И УПРАЖНЕНИЯ

1. Пусть P_{ik} — матрица перестановки. Показать, что вектор $P_{ik}x$ получается из вектора x перестановкой элементов с номерами i, k.

2. Как следствие показать, что матрица $P_{ik}A$ получается из матрицы A перестановкой строк с номерами i, k.

3. Пусть P_{ik} — матрица перестановки. Показать, что $P_{ik}^{-1} = P_{ik}^T = P_{ik}$.

4. Показать, что нижняя треугольная матрица L (с элементами l_{ij}) равна произведению элементарных нижних треугольных матриц L_k (см. (5)), т. е. $L = L_1 L_2 \cdots L_{n-1} L_n$.

Указание. Проведите вычисления в соответствии со следующей расстановкой скобок: $L = L_1(L_2 \cdots (L_{n-2}(L_{n-1}L_n)\cdots)$, т. е. сначала перемножьте $L_{n-1}L_n$, результат умножьте слева на L_{n-2} и т. д.

5. Пусть L_k есть элементарная нижняя треугольная матрица и $l_{kk} \neq 0$. Показать, что

$$L_k^{-1} = \begin{bmatrix} 1 & \dots & 0 & 0 & \dots & 0 \\ \dots & \dots & \dots & \dots & \dots & \dots \\ 0 & \dots & 1/l_{k,k} & 0 & \dots & 0 \\ 0 & \dots & -l_{k+1,k}/l_{k,k} & 1 & \dots & 0 \\ \dots & \dots & \dots & \dots & \dots & \dots \\ 0 & \dots & -l_{n,k}/l_{k,k} & 0 & \dots & 1 \end{bmatrix}.$$

6. Пусть L — нижняя треугольная матрица, у которой все элементы главной диагонали отличны от нуля. Показать, что матрица L^{-1} существует и является нижней треугольной матрицей. Показать, что аналогичное верно и для верхней треугольной матрицы.

7. Доказать, что

a) произведение вектор-столбца x на вектор-строку y есть матрица ранга 1.

b) произведение ортогональных матриц есть ортогональная матрица.

c) если матрица A ортогональна, то ортогональными будут и транспонированная к ней и обратная к A матрицы.

d) произведение матриц перестановок есть также матрица перестановок.

§ 3. Метод исключения Гаусса

В основе метода Гаусса, как, впрочем, и многих других методов решения систем линейных алгебраических уравнений

$$Ax = b, \tag{6}$$

лежит следующее утверждение. Пусть матрица B невырождена. Тогда система уравнений

$$BAx = Bb \tag{7}$$

эквивалентна системе (6), т. е. решение системы (7) — решение системы (6) и, наоборот, решение системы (6) — решение системы (7).

Действительно, пусть x — решение системы (7). Тогда

$$B(Ax - b) = 0,$$

но матрица B невырождена, следовательно, $Ax - b = 0$. Обратное утверждение очевидно.

Матрица B выбирается так, чтобы матрица BA была проще матрицы A и решение системы (7) находилось легче, чем решение системы (6). В методе Гаусса матрица B конструируется при помощи элементарных нижних треугольных матриц так, чтобы матрица BA была верхней треугольной. Тогда решение системы (7) становится тривиальной задачей.

Приведем традиционное описание этого метода, ориентированное на операции со строками A.

1. Расчетные формулы. Для удобства изложения положим $A^{(1)} = A$, $b^{(1)} = b$ и запишем исходную систему в индексной форме:

$$\begin{array}{l} a_{11}^{(1)}x_1 + a_{12}^{(1)}x_2 + \ldots + a_{1n}^{(1)}x_n = b_1^{(1)}, \\ a_{21}^{(1)}x_1 + a_{22}^{(1)}x_2 + \ldots + a_{2n}^{(1)}x_n = b_2^{(1)}, \\ \cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots \\ a_{n1}^{(1)}x_1 + a_{n2}^{(1)}x_2 + \ldots + a_{nn}^{(1)}x_n = b_n^{(1)}. \end{array} \tag{8}$$

Предположим, что $a_{11}^{(1)} \neq 0$ и введем *множители*

$$l_{i1} = a_{i1}^{(1)}/a_{11}^{(1)}, \quad i = 2 : n.$$

Для каждого $i = 2 : n$, умножим обе части первого уравнения в (8) на l_{i1} и вычтем полученное равенство из i-го уравнения. Придем к новой (эквивалентной) системе $A^{(2)}x = b^{(2)}$ вида

$$\begin{array}{r} a_{11}^{(1)}x_1 + a_{12}^{(1)}x_2 + \ldots + a_{1n}^{(1)}x_n = b_1^{(1)}, \\ a_{22}^{(2)}x_2 + \ldots + a_{2n}^{(2)}x_n = b_2^{(2)}, \\ \cdots\cdots\cdots\cdots\cdots\cdots \\ a_{n2}^{(2)}x_2 + \ldots + a_{nn}^{(2)}x_n = b_n^{(2)}. \end{array} \tag{9}$$

Согласно описанию, данному выше, новые элементы матрицы и правой части вычисляются по формулам

$$\begin{array}{l} a_{ij}^{(2)} = a_{ij}^{(1)} - l_{i1}a_{1j}^{(1)}, \quad i, j = 2 : n, \\ b_i^{(2)} = b_i^{(1)} - l_{i1}b_1^{(1)}, \quad i = 2 : n. \end{array} \tag{10}$$

Говорят, что в системе (9) неизвестная x_1 исключена из уравнений со второго по n-е или, что матрица системы приведена к верхней треугольной форме в первом столбце. На этом заканчивается описание первого шага метода Гаусса.

На втором шаге проделаем аналогичные вычисления с подсистемой (9), включающей уравнения с номерами $2:n$, и приведем матрицу системы к верхней треугольной форме во втором столбце. Это можно сделать, если $a_{22}^{(2)} \neq 0$. Повторяя вычисления, получим n систем

$$A^{(k)}x = b^{(k)}, \quad k = 1:n,$$

с матрицами вида

$$A^{(k)} = \begin{bmatrix} a_{11}^{(1)} & a_{12}^{(1)} & \dots & a_{1k}^{(1)} & \dots & a_{1n}^{(1)} \\ 0 & a_{22}^{(2)} & \dots & a_{2k}^{(2)} & \dots & a_{2n}^{(2)} \\ \vdots & & \ddots & & & \vdots \\ 0 & \dots & 0 & a_{kk}^{(k)} & \dots & a_{kn}^{(k)} \\ \vdots & & \vdots & \vdots & & \vdots \\ 0 & \dots & 0 & a_{nk}^{(k)} & \dots & a_{nn}^{(k)} \end{bmatrix}, \quad k \geqslant 1.$$

Ясно, что при $k = n$ (после $n-1$ шага) получим систему $A^{(n)}x = b^{(n)}$ с верхней треугольной матрицей

$$\begin{bmatrix} a_{11}^{(1)} & a_{12}^{(1)} & \dots & a_{1n}^{(1)} \\ 0 & a_{22}^{(2)} & \dots & a_{2n}^{(2)} \\ \vdots & & \ddots & \vdots \\ 0 & & & a_{nn}^{(n)} \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \end{bmatrix} = \begin{bmatrix} b_1^{(1)} \\ b_2^{(2)} \\ \vdots \\ b_n^{(n)} \end{bmatrix}, \tag{11}$$

решая которую получим искомое решение. Переход от исходной системы (8) к системе (11) называется *прямым ходом метода Гаусса*. Решение системы (11) обратной подстановкой — *обратным ходом*. Элементы $a_{ii}^{(i)}$, $i = 1:n$, называются *ведущими (главными) элементами метода Гаусса* и только на них производится деление в ходе вычислений. Для осуществимости метода *они должны быть отличны от нуля*.

Суммируя сказанное, приходим к следующим расчетным формулам. Для всех $k = 1:n-1$ сначала вычисляются множители

$$l_{ik} = a_{ik}^{(k)}/a_{kk}^{(k)}, \quad i = k+1:n. \tag{12}$$

Затем вычисляются новые элементы матрицы $A^{(k+1)}$ и вектора $b^{(k+1)}$:

$$a_{ij}^{(k+1)} = a_{ij}^{(k)} - l_{ik}a_{kj}^{(k)}, \quad i,j = k+1:n, \tag{13}$$
$$b_i^{(k+1)} = b_i^{(k)} - l_{ik}b_k^{(k)}, \quad i = k+1:n.$$

Можно заметить, что при программной реализации этих формул, элементы $a_{ij}^{(k+1)}$ можно хранить на месте элемента a_{ij} исходной матрицы, также как l_{ik} — на месте элемента a_{ik}, $b_i^{(k+1)}$ — на месте b_i.

2. Трудоемкость метода Гаусса. Ясно, что трудоемкость метода Гаусса вычисляется по формуле

$$Q = \sum_{k=1}^{n-1}(q_{mk} + q_{ak} + q_{bk}) + n^2 + n - 1,$$

где q_{mk}, q_{ak}, q_{bk} есть число операций, необходимых для вычисления множителей на шаге k, новых элементов матрицы $A^{(k+1)}$ и вектора $b^{(k+1)}$ по формулам (12), (13) соответственно, а n^2+n-1 есть трудоемкость обратной подстановки.

Используем хорошо известные формулы:

$$1 + 2 + \cdots + m = \frac{m(m+1)}{2},$$
$$1 + 2^2 + \cdots + m^2 = \frac{m(m+1)(2m+1)}{6}.$$

Ясно, что

$$q_m = \sum_{k=1}^{n-1}(n-k) = \sum_{k=1}^{n-1} k = (n-1)n/2.$$

Для вычисления $b^{(k+1)}$ требуется в два раза больше операций, т. е. $q_b = (n-1)n$. Наконец,

$$q_a = \sum_{k=1}^{n-1} 2(n-k)^2 = 2\sum_{k=1}^{n-1} k^2 = (n-1)n(2n-1)/3.$$

Суммарно, $Q = 2n^3/3 + 3n^2/2 - n/6 - 1 = 2n^3/3 + O(n^2)$ flop.

3. Матричная формулировка метода Гаусса. Для $k = 1 : n-1$ определим элементарную треугольную матрицу L_k, где $l_{i,k}$ есть множители (12) метода Гаусса:

$$L_k = \begin{bmatrix} 1 & \cdots & 0 & 0 & \cdots & 0 \\ \cdots & \cdots & \cdots & \cdots & \cdots & \cdots \\ 0 & \cdots & 1 & 0 & \cdots & 0 \\ 0 & \cdots & -l_{k+1,k} & 1 & \cdots & 0 \\ \cdots & \cdots & \cdots & \cdots & \cdots & \cdots \\ 0 & \cdots & -l_{n,k} & 0 & \cdots & 1 \end{bmatrix}.$$

Матрица L_k отличается от единичной только поддиагональными элементами k-го столбца. Непосредственными вычислениями легко проверить (убедитесь!), что система уравнений после первого шага метода Гаусса равносильна системе $L_1 Ax = L_1 b$, т. е. $A^{(2)} = L_1 A$, $b^{(2)} = L_1 b$ (см. формулы (10)). Система уравнений после k-го шага равносильна системе $L_k A^{(k)} x = L_k b^{(k)}$, т.е. $A^{(k+1)} = L_k A^{(k)}$, $b^{(k+1)} = L_k b^{(k)}$ (см. формулы (13)). Обозначим $A^{(n)}$ через U. Тогда

$$U = L_{n-1} L_{n-2} \dots L_1 A, \quad b^{(n)} = L_{n-1} L_{n-2} \dots L_1 b.$$

Отсюда находим

$$A = LU,$$

где $L = L_1^{-1} L_2^{-1} \cdots L_{n-1}^{-1}$. Нетрудно видеть (см. далее задачу 5), что

$$L_k^{-1} = \begin{bmatrix} 1 & \cdots & 0 & 0 & \cdots & 0 \\ \cdots & \cdots & \cdots & \cdots & \cdots & \cdots \\ 0 & \cdots & 1 & 0 & \cdots & 0 \\ 0 & \cdots & l_{k+1,k} & 1 & \cdots & 0 \\ \cdots & \cdots & \cdots & \cdots & \cdots & \cdots \\ 0 & \cdots & l_{n,k} & 0 & \cdots & 1 \end{bmatrix},$$

а матрица L является нижней треугольной с единичной главной диагональю и поддиагональными элементами равными l_{ij} (см. далее задачу 4). Если поддиагональные элементы матрицы L и элементы U хранить на месте соответствующих элементов A, то приходим к следующему алгоритму LU разложения матрицы A (см. расчетные формулы (12), (13)).

```
for k = 1:n-1
  for i = k+1:n
    a(i,k) = a(i,k)/a(k,k);
    for j = k+1:n
      a(i,j) = a(i,j) - a(i,k)*a(k,j);
    end
  end
end
```

Этот алгоритм называется kij – алгоритмом; kji – алгоритм получается перестановкой циклов по i и j.

4. Условия применимости метода Гаусса. Описанный выше метод может быть реализован лишь в том случае, когда все ведущие элементы метода Гаусса отличны от нуля. Для этого невырожденности матрицы недостаточно как показывает следующий пример:

$$A = A^{(1)} = \begin{bmatrix} 1 & 1 & 0 \\ 1 & 1 & 1 \\ 0 & 1 & 1 \end{bmatrix}, \quad \det A = -1, \quad A^{(2)} = \begin{bmatrix} 1 & 1 & 0 \\ 0 & 0 & 1 \\ 0 & 1 & 1 \end{bmatrix}, \quad a_{22}^{(2)} = 0.$$

Выделим класс матриц, для которых метод Гаусса осуществим. Пусть

$$A_1 = a_{11}, \ A_2 = \begin{vmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{vmatrix}, \ \ldots, A_n = \begin{vmatrix} a_{11} & a_{22} & \ldots & a_{1n} \\ a_{21} & a_{22} & \ldots & a_{2n} \\ \ldots & \ldots & \ldots & \ldots \\ a_{n1} & a_{n2} & \ldots & a_{nn} \end{vmatrix}$$

есть главные миноры матрицы A.

Теорема 1. *Для того, чтобы все ведущие элементы метода Гаусса были отличны от нуля необходимо и достаточно, чтобы все главные миноры матрицы A были ненулевыми.*

Доказательство. Напомним, что $a_{ij}^{(1)} = a_{ij}$, $i, j = 1 : n$. Пусть $A_i \neq 0$, $i = 1 : n$. Покажем по индукции, что тогда $a_{kk}^{(k)} \neq 0$ для всех $k = 1 : n$. Имеем, $a_{11}^{(1)} = a_{11} = A_1 \neq 0$. Пусть уже доказано, что $a_{11}^{(1)}, a_{22}^{(2)}, \ldots, a_{k-1,k-1}^{(k-1)}$ не равны нулю. Тогда, приводя минор A_k к треугольному виду при помощи преобразований прямого хода метода Гаусса, получим

$$A_k = \begin{vmatrix} a_{11}^{(1)} & a_{22}^{(1)} & \ldots & a_{1k}^{(1)} \\ 0 & a_{22}^{(2)} & \ldots & a_{2k}^{(2)} \\ \ldots & \ldots & \ldots & \ldots \\ 0 & 0 & \ldots & a_{kk}^{(k)} \end{vmatrix} = a_{11}^{(1)} a_{22}^{(2)} \ldots a_{kk}^{(k)}, \tag{14}$$

следовательно, $a_{kk}^{(k)} \neq 0$, что завершает шаг индукции. Обратное утверждение теоремы есть следствие соотношения (14). $\square$[1)]

Следствием теоремы (1) и разложения $A = LU$ является

Теорема 2. *Пусть все главные миноры матрицы A отличны от нуля. Тогда справедливо единственное представление $A = LU$, где L нижняя треугольная матрица с единичной главной диагональю, U — верхняя треугольная матрица.*

Доказательство. Доказательства требует лишь единственность разложения. Предположим, что имеются два разложения $A = L_1U_1$ и $A = L_2U_2$, т. е. $L_1U_1 = L_2U_2$. Следовательно, $L_2^{-1}L_1 = U_2U_1^{-1}$, причем левая часть этого равенства представляет собой нижнюю треугольную матрицу с единичной диагональю, а правая часть — верхнюю треугольную матрицу. Это возможно только тогда, когда $L_2^{-1}L_1 = I$, $U_2U_1^{-1} = I$, т. е. при $L_1 = L_2$ и $U_1 = U_2$. $\square$

Укажем часто встречающиеся типы матриц, для которых метод Гаусса применим и справедливо разложение $A = LU$.

i) *Симметричные положительно определенные матрицы.* Под $A > 0$ будем понимать, что матрица A положительно определена, т.е.

$$A > 0 \quad \Leftrightarrow \quad (Ax, x) = \sum_{i,j=1}^{n} a_{ij}x_jx_i > 0 \quad \forall\, x \neq 0.$$

В частности, запись $A = A^T > 0$ означает, что A является симметричной положительно определенной матрицей. В соответствии с критерием Сильвестра симметричная матрица положительно определена тогда и только тогда, когда все ее главные миноры положительны.

ii) *Матрицы с диагональным преобладанием.* Матрица, элементы которой удовлетворяют условию:

$$\sum_{j=1,\, j\neq i}^{n} |a_{ij}| < |a_{ii}|, \quad i = 1 : n. \tag{15}$$

называется матрицей с диагональным преобладанием по строкам. Аналогично, матрицы, элементы которой удовлетворяют условию:

$$\sum_{i=1,\, i\neq j}^{n} |a_{ij}| < |a_{jj}|, \quad j = 1 : n.$$

[1)] Значком $\square$ отмечаем конец доказательства.

называются матрицей с диагональным преобладанием по столбцам. Говорят, что A есть матрица диагональным преобладанием, если она имеет диагональное преобладание либо по строкам, либо по столбцам.

Ясно, что если A есть матрица с диагональным преобладанием по строкам, то A^T имеет диагональное преобладание по столбцам.

Теорема 3. *Пусть A есть матрица с диагональным преобладанием. Тогда все ее главные миноры отличны от нуля.*

Доказательство. Достаточно считать, что A имеет диагональное преобладание по строкам. Рассмотрим главный минор A_k, $k \geqslant 1$. Достаточно убедиться, что однородная система линейных уравнений

$$\begin{aligned} &a_{11}x_1 + a_{12}x_2 + \ldots + a_{1k}x_k = 0,\\ &a_{21}x_1 + a_{22}x_2 + \ldots + a_{2k}x_k = 0,\\ &\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\\ &a_{k1}x_1 + a_{k2}x_2 + \ldots + a_{kk}x_k = 0 \end{aligned} \tag{16}$$

с матрицей, составленной из элементов минора A_k, имеет только нулевое решение. Предположим противное и пусть $\max\limits_{1\leqslant j\leqslant k} |x_j| = |x_i|$. Ясно, что $x_i \neq 0$. Поскольку $i \leqslant k$, то из i-е уравнения системы (16) следует

$$a_{ii}x_i = -\sum_{j=1,\, j\neq i}^{k} a_{ij}x_j.$$

Следовательно,

$$|a_{ii}||x_i| \leqslant \sum_{j=1,\, j\neq i}^{k} |a_{ij}|\,|x_j| \leqslant |x_i| \sum_{j=1,\, j\neq i}^{n} |a_{ij}|,$$

т.е. $|a_{ii}| \leqslant \sum_{j=1,\ j\neq i}^{n} |a_{ij}|$, а это противоречит условию (15). $\square$

5. Метод Гаусса с выбором ведущего элемента. Опишем модификацию изученного выше метода Гаусса, который применим для решения систем уравнений с любой невырожденной матрицей.

Выберем среди элементов первого столбца матрицы A максимальный по модулю. Пусть это есть элемент $a_{i_1,1}$. Он не может оказаться равным нулю, так как тогда все элементы первого столбца матрицы A — нули и, значит, $\det(A) = 0$, что противоречит условию $\det(A) \neq 0$.

Умножим обе части уравнения $Ax = b$ на матрицу перестановки $P_{i_1,1}$. В дальнейшем будем обозначать эту матрицу через P_1 (заметим, что она равна единичной, если максимальный по модулю элемент первого столбца матрицы A есть a_{11}). Получим

$$A^{(1)}x = b^{(1)}, \tag{17}$$

где $A^{(1)} = P_1A$, $b^{(1)} = P_1b$. Поясним, что матрица $A^{(1)}$ получается из матрицы A перестановкой первой и i_1-й строк, вектор-столбец $b^{(1)}$ получается из столбца b перестановкой первого и i_1-го элементов. Элементы матрицы $A^{(1)}$ обозначим через $a^{(1)}_{kl}$, элементы столбца $b^{(1)}$ — через $b^{(1)}_k$. По построению $a^{(1)}_{11} \neq 0$.

Теперь можем осуществить первый шаг рассмотренного ранее метода Гаусса и привести матрицу $A^{(1)}$ к верней треугольной форме в первом столбце. Это равносильно умножению обеих частей уравнения (17) на элементарную нижнюю треугольную матрицу L_1. Она была определена при матричной формулировкке метода Гаусса. В результате, придем к системе уравнений

$$A^{(2)}x = b^{(2)}, \tag{18}$$

где $A^{(2)} = L_1A^{(1)} = L_1P_1A$, $b^{(2)} = L_1b^{(1)} = L_1P_1b$. На этом заканчивается первый шаг исключения неизвестных.

На втором шаге среди элементов $a^{(2)}_{22}$, $a^{(2)}_{32}$, $\ldots$, $a^{(2)}_{n2}$ (поддиагональных элементов второго столбца, включая диагональный) найдем максимальный по модулю. Пусть этот элемент есть $a^{(2)}_{i_2,2}$. Он не может равняться нулю. Действительно, если он равен нулю, то все числа $a^{(2)}_{22}$, $a^{(2)}_{32}$, $\ldots$, $a^{(2)}_{n2}$ — нули и тогда, вычисляя $|A^{(2)}|$ разложением по первому столбцу, получим, что $|A^{(2)}| = 0$. С другой стороны, поскольку $|L_1| = 1$, а $|P_1| \neq 0$, то $|A^{(2)}| = |L_1|\,|P_1|\,\det(A) \neq 0$, что приводит к противоречию.

Умножим обе части уравнения (18) на матрицу $P_2 = P_{i_2,2}$, т. е. поменяем местами вторую и i_2-ю строки матрицы $A^{(2)}$. Получим

$$\tilde{A}^{(2)}x = P_2L_1P_1b. \tag{19}$$

По определению элемент $\tilde{a}^{(2)}_{22} \neq 0$. Это позволяет осуществить второй шаг рассмотренного ранее метода Гаусса и привести матрицу $\tilde{A}^{(2)}$ к верней треугольной форме и во втором столбце. Это равносильно

умножению обеих частей уравнения (19) на элементарную нижнюю треугольную матрицу L_2. В результате второго шага получим систему уравнений

$$A^{(3)}x = L_2P_2L_1P_1b,$$

где $A^{(3)} = L_2P_2L_1P_1A$. Продолжая этот процесс, после $n-1$ шага получим систему с верхней треугольной матрицей $U = A^{(n)}$,

$$Ux = f \tag{20}$$

(очевидно, эквивалентную исходной), где

$$U = L_{n-1}P_{n-1}\cdots L_1P_1A, \tag{21}$$

$$f = L_{n-1}P_{n-1}\cdots L_1P_1b.$$

Решение системы (20) не вызывает затруднений.

Замечание 1. Выбор максимального по модулю элемента столбца при выполнении прямого хода метода Гаусса минимизирует влияние ошибок округления. Если не заботиться об ошибках округления, то на очередном шаге прямого хода метода Гаусса можно выбирать любой ненулевой элемент столбца.

Теорема 4. *Пусть* $\det(A) \neq 0$. *Тогда справедливо разложение* $PA = LU$, *где* L *— нижняя треугольная матрица с единичной главной диагональю,* U *— верхняя треугольная матрица,*

$$P = P_{i_{n-1},n-1}P_{i_{n-2},n-2}\cdots P_{i_1,1}$$

— матрица перестановок, $i_k \geqslant k$, $k = 1 : n-1$.

Доказательство. Согласно формуле (21)

$$A = P_1L_1^{-1}\cdots P_{n-2}L_{n-2}^{-1}P_{n-1}L_{n-1}^{-1}U. \tag{22}$$

Здесь мы учли, что произведение P_kP_k есть единичная матрица. Это также позволяет эквивалентно преобразовать (22) к виду

$$P_{n-1}P_{n-2}\cdots P_1A = \Big(P_{n-1}P_{n-2}\cdots P_2L_1^{-1}P_2P_3\cdots P_{n-1}\Big)$$
$$\Big(P_{n-1}\cdots P_3L_2^{-1}P_3\cdots P_{n-1}\Big)\cdots\Big(P_{n-1}L_{n-2}^{-1}P_{n-1}\Big)L_{n-1}^{-1}U =$$
$$= \big(\widetilde{L}_1^{-1}\widetilde{L}_2^{-1}\cdots\widetilde{L}_{n-2}^{-1}L_{n-1}^{-1}\big)U.$$

Отсюда следует утверждение теоремы. Действительно, каждая из матриц $\widetilde{L}_k^{-1}$ представляет собой элементарную нижнюю треугольную матрицу с единичной диагональю, отличающуюся от L_k^{-1} лишь

перестановкой поддиагональных элементов в k-м столбце, а матрица $L = \widetilde{L}_1^{-1} \dots \widetilde{L}_{n-2}^{-1} L_{n-1}^{-1}$ есть нижняя треугольная с единичной диагональю. $\square$

Программная реализации LU разложения матрицы методом Гаусса с выбором ведущего элемента по столбцу осуществляется также, как и описанное ранее LU разложение. Необходимо лишь внести изменения, связанные с перестановкой строк матрицы и запоминанием этих перестановок. Например, kij алгоритм примет вид:

```
function [A,p] = lukij(A)
n = size(A,1);
p = 1:n;
for k = 1:n-1
  [~, I] = max(abs(A(k:n,k)));
  row = I+k-1;
  a([k, row],:) = a([row, k],:);
  p([k, row])   = p([row, k]);
  for i = k+1:n
    a(i,k) = a(i,k)/a(k,k);
    for j = k+1:n
      a(i,j) = a(i,j) - a(i,k)*a(k,j);
    end
  end
end
```

В результате выполнения этой функции, матрицы L и U сохраняются на месте матрицы A. В векторе p сохраняются перестановки строк.

Пусть $[LU, p] = lukij(A)$. Тогда команды $L = tril(LU, -1) + eye(n)$; $U = triu(LU)$ позволяют при необходимости получить L и U. Вектор перестановок p таков, что $A(p, :) = LU$.

Задания для самостоятельной работы

ВОПРОСЫ ДЛЯ САМОКОНТРОЛЯ

1. В чем заключается идея метода исключения Гаусса?

2. Приведите расчетные формулы метода исключения Гаусса.

3. Какие элементы метода исключения Гаусса называются ведущими?

4. Применим ли метод Гаусса к системам a) с невырожденной матрицей? b) ортогональной матрицей?

5. Сформулируйте условия применимости метода исключения Гаусса.

6. Укажите трудоемкость метода исключения Гаусса.

7. Пусть система из $n \gg 1$ уравнений решается методом Гаусса за 1 единицу времени. Примерно какое число единиц времени потребует решение системы размера $10n$?

8. Для каких матриц A справедливо разложение $A = LU$?

9. Для каких матриц A справедливо разложение $PA = LU$, где P есть матрица перестановок?

10. В чем отличие метода Гаусса и метода Гаусса с выбором ведущего элемента?

ЗАДАЧИ И УПРАЖНЕНИЯ

1. Пусть диагональные элементы L и U равны единице. Получить формулы для элементов L^{-1} и U^{-1}. Оценить трудоемкость.

2. Пусть M есть множество симметричных матриц размера 2×2 с элементами из отрезка $[0, 1]$, которые можно представить в ЭВМ в плавающей системе double. Определить вероятность того, что СЛАУ со случайно выбранной матрицей $A \in M$ разрешима.

Указание. Число типа double в памяти ЭВМ занимает 64 разряда: 1 бит для знака числа, 11 — для хранения экспоненты и 52 — для мантиссы. Следовательно, числа типа double из интервала [0,1] имеют вид $0.d_1d_2...d_{52}$, где d_k равны 0 или 1, т.е. они образуют равномерную сетку на отрезке [0,1] с шагом $\varepsilon = 2^{-52} \approx 2.2 \cdot 10^{-16}$ (относительная погрешность представления чисел в ЭВМ или машинный ипсилон).

3. Пусть A — кососимметричная вещественная матрица, т.е. $A = -A^T$. Найти те значения α, при которых матрица $A + \alpha I$ обратима (I — единичная матрица) в случае
   a) когда порядок A — нечетное число;
   b) когда порядок A — четное число.

4. Пусть известно LU разложение матрицы A. За сколько арифметических операций можно решить N систем $Ax_k = b_k$, $k = 1 : N$, в этом случае?

Указание. Заметьте, что единичная задача сводится к двум треугольным системам: $Ly = b_k$, $Ux_k = y$.

5. Пусть матрица A симметрична и ее главные миноры отличны от нуля. Докажите, что существует единственное разложение $A = LDL^T$, называемое LDL разложением матрицы A. Здесь L — нижняя треугольная матрица с единичной диагональю, D — диагональная матрица.

6. Пусть матрица A симметричная и ее главные миноры отличны от нуля. Получите расчетные формулы для вычисления элементов L и D в разложении $A = LDL^T$.

Указание. Действуйте также, как и при выводе формул компактного метода Гаусса. В силу симметрии достаточно рассмотреть случаи $i > j$ и $i = j$ (или $i < j$ и $i = j$).

§ 4. Компактные схемы метода Гаусса

Предположим, что LU разложение матрицы A существует. Рассмотрим другие способы его получения, отличные от метода Гаусса.

1. *LU* разложения матрицы. Посмотрим на разложение $A = LU$ как на уравнение $LU = A$ для определения элементов матриц L и U. Тогда получим n^2 уравнений

$$\sum_{k=1}^{n} l_{ik}u_{kj} = a_{ij}, \quad i, j = 1 : n. \tag{23}$$

Поскольку $l_{ii} = 1$, $l_{ik} = 0$, если $k > i$, и $u_{kj} = 0$, если $k > j$, то равенства (23) можно записать в виде

$$\sum_{k=1}^{\min(i,j)} l_{ik}u_{kj} = a_{ij}\,, \quad i,j = 1:n. \tag{24}$$

Рассмотрим два случая: $i > j$ и $i \leqslant j$.

1) При $i > j$ в формуле (24) верхний индекс суммирования равен j. Тогда равенства запишутся в виде

$$\sum_{k=1}^{j-1} l_{ik}u_{kj} + l_{ij}u_{jj} = a_{ij}\,, \quad i > j\,.$$

Отсюда следует, что

$$l_{ij} = \Big(a_{ij} - \sum_{k=1}^{j-1} l_{ik}u_{kj}\Big)/u_{jj}\,, \quad i > j\,. \tag{25}$$

2) При $i \leqslant j$ из (24) получим

$$\sum_{k=1}^{i-1} l_{ik}u_{kj} + u_{ii} = a_{ij}\,, \quad i \leqslant j\,,$$

откуда вытекают формулы

$$u_{ij} = a_{ij} - \sum_{k=1}^{i-1} l_{ik}u_{kj}\,, \quad i \leqslant j\,. \tag{26}$$

Из формул (25), (26) можно получить различные алгоритмы вычисления элементов L и U, если определить порядок их вычисления.

Например, следующий ijk-алгоритм позволяет вычислять элементы L и U построчно: для всех $i = 1:n$, сначала вычисляются l_{ij} по формулам (25) для всех $j = 1:i-1$, а затем u_{ij} по формулам (26) для всех $j = i:n$ (проверьте!).

В jik-алгоритме элементы L и U вычисляются по столбцам: для всех $j = 1:n$, сначала вычисляются u_{ij} по формулам (26) для всех $i = 1:j$, а затем l_{ij} по формулам (25) для всех $i = j+1:n$ (поверьте!).

Вычисления, аналогичные методу Гаусса, показывают, что трудоемкость этих методов одна и та же и равна $(2/3)\,n^3 + O(n^2)$ flop. Отметим также, что как и в рассмотренном первоначально методе LU разложения, элементы L и U в ходе вычисления можно располагать в соответствующих позициях матрицы A.

2. LDL разложение. Метод Холецкого. Если матрица системы линейных уравнений симметрична и положительно определена, можно добиться двукратного сокращения числа операций и памяти, необходимых для разложения ее на треугольные множители. В основе соответствующего метода лежит

Теорема 5. *Если матрица A симметрична и положительно определена, то справедливы следующие разложения, в которых матрицы определяются однзначно:*

1) $A = LDL^T$, где L — нижняя треугольная матрица с единичной главной диагональю, а D — диагональная матрица с положительными элементами (LDL разложение A);

2) $A = CC^T$, где C — нижняя треугольная матрица с положительной главной диагональю (разложение Холецкого A)[1)].

Доказательство. Поскольку все главные миноры A положительны, то существует единственное треугольное разложение $A = LU$, причем на главной диагонали L стоят единицы, а на главной диагонали U — ведущие элементы метода исключения Гаусса, отличные от нуля.

Пусть $D = \operatorname{diag}(u_{11}, \ldots, u_{nn})$ — диагональная матрица, образованная диагональными элементами U. Определим матрицу $\widetilde{U} = D^{-1}U$, на главной диагонали которой стоят единицы. Тогда $U = D\widetilde{U}$ и $A = LD\widetilde{U}$. Из симметрии матрицы следует, что $\widetilde{U} = L^T$. Следовательно, $A = LDL^T$. Пусть $e_i = (0, \ldots, 0, 1, 0, \ldots, 0)^T$ — орт i-той координатной оси, $e = (L^T)^{-1}e_i$. Тогда

$$0 < (Ae, e) = (LDL^Te, e) = (DL^Te, L^Te) = (De_i, e_i) = u_{ii}.$$

Это доказывает 1). Определим матрицу $D^{1/2} = \operatorname{diag}(u_{11}^{1/2}, \ldots, u_{nn}^{1/2})$. Ясно, что $D^{1/2}D^{1/2} = D$. Тогда $A = LD^{1/2}D^{1/2}L^T = CC^T$, где $C = LD^{1/2}$. Легко видеть, что главные диагонали C и $D^{1/2}$ совпадают. □

Замечание 2. 1) В случае симметричной матрицы в памяти ЭВМ можно хранить только нижнюю треугольную (или верхнюю треугольную) часть A. Это дает практически двукратную экономию памяти ЭВМ, что существенно при больших n.

2) Если известно разложение $A = LDL^T$, то решение системы уравнений $Ax = b$ может быть найдено как последовательное решение трех систем: $Lz = b$, $Dy = z$, $L^Tx = y$. Это потребует $2n^2 + O(n)$ flop (проверьте!)

3) Если известно разложение $A = CC^T$, то решение $Ax = b$ находится как последовательное решение двух треугольных систем: $Cy = b$, $C^Tx = y$. Это потребует также $2n^2 + O(n)$ flop (проверьте!)

[1)] Андре-Луи Холецкий (Andre-Louis Cholesky; 1875-1918) — французский математик

4) Можно сделать предположение, что метод Холецкого в два раза экономичнее метода Гаусса, поскольку вместо двух треугольных матриц L и U в разложении $A = LU$, требуется определить только одну — C. Это же верно и для LDL разложения.

3. LDL разложение. По аналогии с компактной схемой LU разложения получим расчетные формулы для LDL разложения. Будем искать элементы l_{ij} матрицы L и диагональные элементы d_i матрицы D, решая систему (см. упр. 4)

$$LDL^T = A \quad \Leftrightarrow \quad \sum_{k=1}^{n} l_{ik} d_k l_{jk} = a_{ij}\,, \quad i, j = 1 : n.$$

Поскольку $l_{ii} = 1$, $l_{ik} = 0$, если $k > i$, с учетом симметрии получим

$$\sum_{k=1}^{j} l_{ik} d_k l_{jk} = a_{ij}\,, \quad i \geqslant j. \tag{27}$$

1) Рассмотрим случай $i = j$. Тогда из (27) получаем

$$d_j = a_{jj} - \sum_{k=1}^{j-1} l_{jk}^2 d_k. \tag{28}$$

2) При $i > j$ из (27) следует

$$l_{ij} d_j + \sum_{k=1}^{j-1} l_{ik} d_k l_{jk} = a_{ij}.$$

Следовательно,

$$l_{ij} = \Big(a_{ij} - \sum_{k=1}^{j-1} l_{ik} d_k l_{jk}\Big)/d_j, \quad i > j. \tag{29}$$

Эти формулы приводят к следующему методу вычисления элементов L по столбцам: для $j = 1 : n$, сначала вычисляются d_j по формулам (28), а затем l_{ij} по формулам (29) для всех $i = j + 1 : n$ (проверьте!). Аналогично получаются строчно ориентированные формулы (см. упр. 6).

4. Схема внешних произведений метода Холецкого. Используем индукцию по порядку матрицы. Для матрицы первого порядка имеем равенство $a_{11} = \sqrt{a_{11}}\sqrt{a_{11}}$. Пусть утверждение теоремы

верно для матриц порядка $k > 1$. Покажем, что тогда оно верно и для матриц порядка $k+1$. Запишем матрицу A_{k+1} порядка $k+1$ как блочную:

$$A_{k+1} = \begin{bmatrix} A_k & a_k \\ a_k^T & a_{k+1,k+1} \end{bmatrix}. \tag{30}$$

Здесь A_k — матрица порядка k. В силу предположения индукции она симметрична и положительно определена, $A_k = C_k C_k^T$, где C_k — нижняя треугольная матрица с положительными элементами на диагонали. Будем искать разложение A_{k+1} на треугольные множители в виде

$$A_{k+1} = C_{k+1} C_{k+1}^T = \begin{bmatrix} C_k & 0 \\ c_k^T & c_{k+1,k+1} \end{bmatrix} \begin{bmatrix} C_k^T & c_k \\ 0 & c_{k+1,k+1} \end{bmatrix}, \tag{31}$$

где вектор столбец c_k длины $n-1$ и число $c_{k+1,k+1}$ подлежат определению. Выполним умножение в правой части (31), учитывая, что умножение блочных матриц осуществляется по тому же правилу, что и числовых матриц. Получим

$$A_{k+1} = \begin{bmatrix} C_k C_k^T & C_k c_k \\ c_k^T C_k^T & c_k^T c_k + c_{k+1,k+1}^2 \end{bmatrix}. \tag{32}$$

Отметим, что $c_k^T C_k^T = (C_k c_k)^T$. Сравнивая поблочно результат с матрицей A_{k+1} (т.е. формулы (30) и (32)), получим систему линейных уравнений

$$C_k c_k = a_k \tag{33}$$

для определения вектора c_k и уравнение $c_k^T c_k + c_{k+1,k+1}^2 = a_{k+1,k+1}$ для элемента $c_{k+1,k+1}$.

Таким образом, для построения матрицы C, начиная с $c_{11} = \sqrt{a_{11}}$, нужно для всех $k = 2 : n$ решить систему (33) с треугольной матрицей, а затем вычислить $c_{k+1,k+1} = \sqrt{a_{k+1,k+1} - c_k^T c_k}$. Отметим, что $a^T a = (a,a) = |a|^2$ — квадрат длины вектора a.

Этот строчно ориентированный метод вычислений называется схемой внешних произведений. Нетрудно видеть, что его реализация по затратам памяти и объему вычислений действительно оказывается примерно в два раза более экономичной, чем разложение на треугольные множители произвольной невырожденной матрицы (см. упр. 2). В упр. 7 указан другой метод получения расчетных формул.

Задания для самостоятельной работы

ВОПРОСЫ ДЛЯ САМОКОНТРОЛЯ

1. В чем заключается идея компактных схем метода Гаусса?

2. Приведите расчетные формулы компактной схемы LU-разложения.

3. Приведите расчетные формулы компактной схемы LDL-разложения.

4. Приведите расчетные формулы метода Холецкого.

5. Пусть разложение $A = LU$ получено методом Гаусса, а разложение $A = L_1U_1$ — компактным методом Гаусса. Совпадают ли матрицы L и L_1, а также U и U_1?

6. Укажите трудоемкость метода Холецкого и сравните ее с трудоемкостью метода Гаусса. Чем объясняется различие?

7. Сравните трудоемкость LDL и LU разложения матрицы. Чем объясняется различие?

ЗАДАЧИ И УПРАЖНЕНИЯ

1. Докажите, что элементы главной диагонали положительно определенной матрицы положительны.

2. Покажите, что трудоемкость метода Холецкого равна $n^3/3 + O(n^2)$ flop.

3. Докажите, что при выполнении условий теоремы 5 нижняя треугольная матрица L в разложении $A = LL^T$ определяется однозначно.

4. Докажите, что из разлжения (31) следует $|A| = |A_k|\, l^2_{k+1,k+1}$.

5. Покажите, что равенство $LDL^T = A$ (см. (27)) равносильно системе

$$\sum_{k=1}^{\min(i,j)} l_{ik} d_k l_{jk} = a_{ij}\,, \quad i \geqslant j.$$

6. Получите строчно ориентированные расчетные формулы LDL разложения.

7. Получите столбцово ориентированные расчетные формулы разложения Холецкого.

Указание. Модифицируйте рассуждения вывода формул LDL разложения матрицы, приведенные выше.

§ 5. Элементарные ортогональные матрицы

Выше мы изучили метод Гаусса и его варианты, эквивалентные разложениям матрицы на треугольные множители. Далее рассмотрим методы, приводящие или основанные на представлении матрицы в виде произведения

$$A = QR, \tag{34}$$

где Q — ортогональная матрица, а R — верхняя треугольная матрица.

Если разложение (34) получено, то решение системы уравнений $Ax = b$ с невырожденной матрицей A сводится к вычислению

вектора $f = Q^T b$ и решению системы уравнений $Rx = f$ с треугольной невырожденной матрицей (проверьте!).

Напомним, что при ортогональном преобразовании длина вектора не меняется, т.к. $|Qx|^2 = (Qx, Qx) = (x, Q^T Qx) = (x, x) = |x|^2$.

При построении разложения (34) используются специальные ортогональные матрицы, позволяющие решить следующую задачу.

Задача 1. *Даны ненулевой вектор* $a \in \mathbb{R}^n$ *и вектор* $e_1 = (1, 0, \ldots, 0)^T$. *Требуется построить ортогональную матрицу* V *такую, что* $Va = \mu\, e_1$, *где* μ — *число (ясно, что* $|\mu| = |a|$*).*

1. Матрицы вращения. Определение 1. *Вещественная матрица* $G = G_{kl} = \{g_{ij}\}_{i,j=1}^n$, $1 \leqslant k < l \leqslant n$, *называется матрицей вращения, если* $g_{ii} = 1$ *при* $i \neq k, l$, $g_{kk} = c$, $g_{ll} = c$, $g_{kl} = -s$, $g_{lk} = s$, *все остальные элементы матрицы* G_{kl} *равны нулю, причем* $|c|^2 + |s|^2 = 1$. *Она имеет вид*

$$G = \begin{pmatrix} 1 & & & & & & & & \\ & 1 & & & & & & & \\ & & \ddots & & & & & & \\ & & & c & & -s & & & \\ & & & & \ddots & & & & \\ & & & s & & c & & & \\ & & & & & & \ddots & & \\ & & & & & & & 1 & \\ & & & & & & & & 1 \end{pmatrix}.$$

Нетрудно видеть, что G — ортогональная матрица, поскольку ее строки и столбцы образуют ортонормированные системы векторов. Ясно также, что матрицу вращения не надо хранить в ЭВМ в виде квадратной матрицы, достаточно хранить четыре числа (k, l, c, s).

Порождаемое матрицей G преобразование евклидова пространства $\mathbb{R}^n$ в себя представляет собой поворот на угол $\varphi = \operatorname{arctg}(s/c)$ в двумерном подпространстве (плоскости), натянутом на векторы e_k, e_l естественного базиса пространства $\mathbb{R}^n$. Матрица $G^T = G^{-1}$ выполняет поворот в той же плоскости в обратном направлении.

Пусть $a \in \mathbb{R}^n$. Ясно, что $(Ga)_i = a_i$ при $i \neq k, l$, и

$$(Ga)_k = c\,a_k - s\,a_l,$$
$$(Ga)_l = s\,a_k + c\,a_l.$$

Как видим умножение G на заданный вектор a требует 6 flop. Пусть $\rho = (|a_k|^2 + |a_l|^2)^{1/2}$. Положим $c = 1, s = 0$, если $\rho = 0$, и $c = a_k/\rho$, $s = -a_l/\rho$, если $\rho > 0$. Тогда $(Ga)_k = \rho$, $(Ga)_l = 0$.

Следовательно, если $a = (a_1, \ldots, a_n)^T \neq 0$ произвольный вектор, то вектор $a^{(1)} = G_{1n}a$ будет равен $a^{(1)} = (\rho_1, a_2, \ldots, a_{n-1}, 0)^T$, где ρ_1 было определено выше при $k = 1$, $l = n - 1$. Это требует 4 бинарных арифметических операций и одно извлечение корня. Далее, аналогично, $a^{(2)} = G_{1,n-1}a^{(1)} = (\rho_2, a_2, \ldots, a_{n-2}, 0, 0)^T$ также за 4 бинарных арифметических операций и одно извлечение корня. Процесс продолжим, пока не получим вектор $a^{(n-1)} = |a|\,e_1$.

Итак, мы получили следующий **метод решения задачи 1**: если $a \neq 0$ — произвольный вектор из $\mathbb{R}^n$, то за $4(n-1)$ flop и $n-1$ извлечения корня, можно определить пары чисел (c_{n-1}, s_{n-1}), (c_{n-2}, s_{n-2}), $\ldots$, (c_1, s_1) и соответствующие им матрицы вращения $G_{1,n}$, $G_{1,n-1}$, $\ldots$, $G_{1,2}$ такие, что $Ga = |a|\,e_1$, где $G = G_{1,2} \cdots G_{1,n-1}G_{1,n}$ — ортогональная матрица (см. упр. 8). Умножение этого G вектор реализуется за $6(n-1)$ flop.

Таким образом, любой ненулевой вектор при помощи ортогональной матрицы можно преобразовать в вектор, совпадающий по направлению с вектором e_1 естественного базиса.

Замечание 3. Пусть теперь a, b — два произвольных ненулевых вектора пространства $\mathbb{R}^n$. Как только что было показано, существуют ортогональные матрицы $G(a)$ и $G(b)$ такие, что $G(a)a = |a|\,e_1$, $G(b)b = |b|\,e_1$. Отсюда вытекает, что $Ga = \mu\,b$, где $\mu = |a|/|b|$, $G = G^T(b)G(a)$, т.е. для любой пары ненулевых векторов найдется ортогональная матрица, преобразующая первый вектор в вектор, совпадающий по направлению со вторым.

2. Матрицы отражения. Пусть произвольно задан вектор $w = (w_1, w_2, \ldots, w_n\}^T$ единичной длины (матрица $n \times 1$). Матрица

$$H = H(w) = I - 2ww^T = \{\delta_{ij} - 2w_iw_j\}_{i,j=1}^n \tag{35}$$

называется *матрицей отражения*. Отметим ряд ее свойств.

1. Матрица H симметрична (что очевидно) и ортогональна, т.к.

$$H^TH = H^2 = I - 4ww^T + 4w(w^Tw)w^T = I,$$

поскольку $w^T w = |w|^2 = 1$. Таким образом, $H = H^T = H^{-1}$.

2. Пусть $E_{n-1} = \{z \in \mathbb{R}^n : w^T z = (z, w) = 0\}$ — гиперплоскость размерности $n-1$, нормальная к вектору w. Заметим, что

$$Hw = w - 2ww^T w = -w, \quad Hz = z - 2ww^T z = z, \quad z \in E_{n-1}. \quad (36)$$

Следовательно, H имеет однократное собственное значение равное -1, которому соответствует собственный вектор w, и собственное значение $+1$ кратности $n-1$, которому соответствует собственное подпространство E_{n-1}. Отсюда следует, что $\det(H) = -1$ [1)].

3. Пусть x — произвольный вектор, а z его проекция на гиперплоскость E_{n-1}. Ясно, что векторы x, z и w лежат в двумерной плоскости, нормальной к E_{n-1}, и x однозначно представим в виде $x = \alpha w + z$, где α некоторое число. Из равенств (36) вытекает, что $Hx = -\alpha w + z$ (сделайте рисунок!). Можно сказать, таким образом, что отображение, порождаемое матрицей H, выполняет отражение вектора x относительно гиперплоскости E_{n-1}, ортогональной вектору w. Это свойство матрицы H и позволяет называть ее матрицей отражения.

4. Произведение $y = H(w)a$ вычисляется по формуле

$$y = (I - 2ww^T)a = a - \lambda\, w, \quad \lambda = 2\, w^T a,$$

а его трудоемкость равна $4n$ flop (убедитесь в этом!).

5. Пусть заданы векторы a, $e \in \mathbb{R}^n$, $|a| \neq 0$, $|e| = 1$. Рассмотрим задачу построения такой матрицы отражения $H = H(w)$, что $Ha = \mu e$, где $|\mu| = |a|$ (см. замечание 3). Из геометрических соображений ясно, что эта задача имеет два решения[2)]. Положим

$$w = (a - \mu e)/\nu, \quad \nu = |a - \mu e|\,.$$

Имеем

$$\nu^2 = (a - \mu e, a - \mu e) = 2\,(a, a - \mu e)\,, \quad (37)$$

$$H(w)a = a - \frac{2(a, a - \mu e)}{\nu^2}\,(a - \mu e). \quad (38)$$

Из формул (37), (38) следует, что $H(w)a = \mu e$.

[1)] Определитель матрицы равен произведению ее собственных чисел

[2)] Проиллюстрируйте построение вектора w рисунком в двумерном случае.

6. Рассмотрим случай, когда $e = e_1 = (1, 0, \ldots, 0)^T$. Тогда $(a, \mu e) = \mu a_1$. Положим $\mu = \pm|a|a_1/|a_1|$, если $a_1 \neq 0$, иначе примем $\mu = \pm|a|$. Итак, решение задачи $H(w)a = \mu e_1$ определяется формулой (35) при

$$w = \frac{v}{|v|}, \quad v = (a_1 - \mu, a_2, \ldots, a_n)^T, \quad \mu = \pm \begin{cases} |a|, & a_1 = 0, \\ \frac{|a|\, a_1}{|a_1|}, & a_1 \neq 0. \end{cases} \tag{39}$$

Конкретное решение (т. е. знак μ) выбирается из дополнительных соображений, например, с целью получить более устойчивый к погрешностям округления метод при вычислениях на ЭВМ.

Экономное вычисление w по формулам (39) требует $3n$ flop и одно извлечение корня. Матрицу отражения в памяти ЭВМ можно не хранить; достаточно хранить только вектор w.

Итак, мы получили два способа решения задачи 1: на основе матриц вращения и отражения. Оба метода требуют $O(n)$ flop для определения матрицы V, но матрицы отражения оказались более экономичными.

§ 6. QR разложение матриц

Теорема 6. *Пусть A — вещественная квадратная матрица. Тогда существует ортогональная матрица Q такая, что*

$$A = QR,$$

где R — верхняя треугольная матрица.

Доказательство. Доказательство является конструктивным и дает метод построения матриц Q, R, называемый *методом отражения.* Он состоит из $n - 1$ шага и на k-м шаге матрица A преобразуется к матрице, имеющей верхнюю треугольную форму в k-м столбце. Обозначим через I_k единичную матрицу размера k.

Пусть a_j есть j-й столбец A. Если $a_1 = 0$, то перейдем ко второму шагу, полагая $H^{(1)} = I_n$, $A^{(1)} = A$. Иначе, выберем $H^{(1)} = H_1(w_1)$ как такую матрицу отражения, что $H^{(1)}a_1 = \mu_1\, e_1$ и вычислим $A^{(1)} = H^{(1)}A$. По определению

$$A^{(1)} = [H^{(1)}a_1,\ H^{(1)}a_2, \ldots, H^{(1)}a_n]\,.$$

На этом заканчивается первый шаг. Матрица $A^{(1)}$ имеет верхнюю треугольную форму в первом столбце и в блочном виде имеет вид

$$A^{(1)} = \begin{bmatrix} \mu_1 & c_1 \\ 0 & A_1 \end{bmatrix},$$

где $\mu_1 = \pm|a_1|a_{11}/|a_{11}|$, если $a_{11} \neq 0$, в противном случае $\mu_1 = \pm|a_1|$, A_1 — некоторая квадратная матрица размера $n-1$.

Подсчитаем трудоемкость. На вычисление w_1 требуется $3n$ flop. Вычисление произведений $H_1(w_1)a_2, \ldots, H_1(w_1)a_n$ требует $4n(n-1)$ flop. Т.о. трудоемкость первого шага равна $4n^2 - n$ flop, если $a_1 \neq 0$.

Аналогично осуществляется второй шаг с той лишь разницей, что вычисления производятся с матрицей A_1. А именно, если первый столбец A_1 равен нулю, положим $H^{(2)} = I_n$, $A^{(2)} = A^{(1)}$. Иначе, определим $A^{(2)} = H^{(2)}A^{(1)}$, где матрица $H^{(2)}$ имеет вид

$$H^{(2)} = \begin{bmatrix} 1 & 0 \\ 0 & H_2(w_2) \end{bmatrix}.$$

В этом случае

$$A^{(2)} = \begin{bmatrix} 1 & 0 \\ 0 & H_2(w_2) \end{bmatrix} \begin{bmatrix} \mu_1 & c_1 \\ 0 & A_1 \end{bmatrix} = \begin{bmatrix} \mu_1 & c_1 \\ 0 & H_2(w_2)A_1 \end{bmatrix}.$$

Как и на первом шаге, выберем матрицу $H_2(w_2)$ как такую матрицу отражения, что $H_2(w_2)A_1 = \mu_2\bar{e}_1$, где $\bar{e}_1 = (1, 0, \ldots, 0) \in \mathbb{R}^{n-1}$. Размерность этой задачи на единицу меньше, чем на первом шаге, и равна $n-1$. Соответственно, трудоемкость второго шага не превосходит $4(n-1)^2 - (n-1)$ flop. Таким образом, $A^{(2)}$ имеет верхнюю треугольную форму в первых двух столбцах. Легко видеть, что матрица $H^{(2)}$ является ортогональной (см. упр. 5).

Повторяя построения на k шаге определим матрицу $A^{(k)} = H^{(k)}A^{(k-1)}$, где

$$H^{(k)} = \begin{bmatrix} I_{k-1} & 0 \\ 0 & H_k(w_k) \end{bmatrix}, \tag{40}$$

если матрица $A^{(k-1)}$ не имеет верхней треугольной формы в k-м столбце (иначе, полагаем $H^{(k)} = I_n$). Матрица $H_k(w_k)$ размера $n-k+1$ строится как соответствующая матрица отражения.

После $n-1$ шага получим матрицы отражения $H^{(1)}$, …, $H^{(n-1)}$ такие, что $H^{(n-1)}H^{(n-2)}\cdots H^{(1)}A = A^{(n-1)} = R$, где R — верхняя

треугольная матрица с диагональными элементми μ_i. Следовательно, $A = QR$, где $Q = H^{(1)}H^{(2)}\cdots H^{(n-1)}$ суть ортогональная матрица.

Трудоемкость метода равна

$$\sum_{k=2}^{n}\Big(4k^2 - k\Big) = \frac{4}{3}\,n^3 + O(n^2),$$

что при больших значениях n в два раз больше, чем требуется для разложения $PA = LU$ методом Гаусса. $\square$

Важным положительным качеством описанного метода является его применимость для произвольной матрицы без какой-либо перенумерации ее строк, а также его устойчивость к ошибкам округления. Последнее объясняется тем, что при ортогональном преобразовании длина вектора не меняется.

§ 7. Вычисление определителей и обратной матрицы

Факторизация матриц полезна при решении различных задач.

1. Вычисление определителя. Пусть A произвольная невырожденная квадратная матрица размера n. Используя метод Гаусса с выбором ведущего элемента по столбцу за $2n^3/3 + O(n^2)$ flop получим разложение $PA = LU$, где $P = P_{i_{n-1},n-1}P_{i_{n-2},n-2}\cdots P_{i_1,1}$ — матрица перестановок. Тогда

$$\det(P)\,\det(A) = \det(L)\,\det(U) = \det(U) = u_{11}u_{22}\ldots u_{nn}.$$

Поскольку, элементарная матрица перестановок $P_{i_k,k}$ получается из единичной перестановкой строк с номерами k и i_k, то $\det(P_{i_k,k}) = -1$, если на k-том шаге перестановка была, иначе $\det(P_{i_k,k}) = 1$. Таким образом, $\det(P) = (-1)^m$, где m число перестановок строк, совершенных в процессе исключения неизвестных. Окончательно получаем,

$$\det(A) = (-1)^m u_{11}u_{22}\ldots u_{nn}.$$

Аналогичные формулы нетрудно написать и в тех случаях, когда строится разложение матрицы на простые множители.

Надо, однако, иметь в виду, что непосредственное вычисление по этой формуле, как правило, оказываются невозможным: из-за большого числа сомножителей определитель (или результат промежуточных вычислений) зачастую либо слишком велик, либо, наоборот,

слишком мал. Приходится использовать специальные алгоритмы раздельного вычисления мантиссы и порядка определителя.

Примеры: $A = \mathrm{diag}(1:n)$, $A = \mathrm{diag}(1, 1/2, \ldots, 1/n)$.

Заметим, что трудоемкость операций вычисления $\det(A)$ и решения СЛАУ $Ax = b$ имеет один и тот же порядок $2n^3/3 + O(n^2)$ flop.

2. Вычисление обратной матрицы. Задача построения обратной матрицы сводится к решению n систем линейных уравнений с одной и той же матрицей A и различными правыми частями. Действительно, обозначим матрицу A^{-1} через $Y = [y_1, y_2, \ldots, y_n]$, где y_j — столбцы Y. Тогда $AY = I$, где I — единичная матрица. Столбцами I являются единичные орты e_j. Поэтому

$$AY = [Ay_1, Ay_2, \ldots, Ay_n] = [e_1, e_2, \ldots, e_n] = I.$$

Отсюда следуют n равенств

$$Ay_k = e_k, \quad k = 1:n.$$

Рассмотрим два способа вычисления обратной матрицы.

1. Методом Гаусса с выбором ведущего элемента по столбцу вычислим матрицу перестановок P и треугольные матрицы L и U такие, что $PA = LU$. Это потребует $2/3n^3 + O(n^2)$ flop. Тогда получаем $LUy_k = p^k$, где $p^k = P^T e_k$ есть k-й столбец P^T. Нахождение y_k требует решения систем $Ly = p^k$, $Ux^k = y$. Их суммарная трудоемкость равна $2n^2 + O(n)$ flop. Следовательно, матрица A^{-1} этим методом вычисляется за $(2 + 2/3)n^3 + O(n^2) = 8/3n^3 + O(n^2)$ flop.

2. Методом отражения найдем разложение $A = QR$, затратив $4/3n^3 + O(n^2)$ flop. Тогда $RY = Q^T$. Определение Y из этого уравнения потребует $n^3 + O(n^2)$ flop. Суммарно, матрица A^{-1} этим методом вычисляется за $(1 + 4/3)n^3 + O(n^2)$ flop, что на $n^3/3$ flop меньше, чем в первом методе при больших n.

Задания для самостоятельной работы

ВОПРОСЫ ДЛЯ САМОКОНТРОЛЯ

1. Дайте определение матрицы вращения. Почему она так назвается?

2. Укажите способ хранения в ЭВМ матрицы вращения.

3. За сколько вращений произвольный вектор можно отразить в вектор, колинеарный данному?

4. Дайте определение матрицы отражения. Почему она так назвается?

5. Укажите способ хранения в ЭВМ матрицы отражения.

6. Сколько арифметических операций требует умножение матрицы отражения на заданный вектор?

7. Чему равен определитель матрицы отражения?

8. Какова трудоемкость метода отражения? Сравните с трудоемкостью метода Гаусса.

9. При больших значениях n выше трудоемкость вычисления определителя матрицы или трудоемкость решения СЛАУ?

10. Каким методом выгоднее вычислять обратную матрицу: на основе LU или QR разложения матрицы?

ЗАДАЧИ И УПРАЖНЕНИЯ

1. Докажите, что произведение матриц отражения есть ортогональная матрица. Найдите определитель произведения матриц отражения.

2. Постройте метод, аналогичный описанному при доказательстве теоремы 6 и основанный на использовании матриц вращения. Оцените трудоемкость.

3. Докажите, что если матрица A невырождена, а диагональные элементы матрицы R считаются положительными, то матрицы Q, R в разложении $A = QR$ определяются однозначно.

4. Укажите метод построения разложений $A = QL$, где Q — ортогональная, L — нижняя треугольная матрицы.

Указание. На первом шаге приведите матрицу к нижнему треугольному виду в последнем столбце и аналогично — на следующих шагах.

5. Докажите, что матрица, определяемая формулой (40) является матрицей отражения и $\det(H^{(k)}) = -1$.

6. Найдите определитель матрицы вращения.

§ 8. Решение разреженных систем уравнений

Кратко рассмотрим вопросы, возникающие при решении СЛАУ $Ax = b$ с разреженной матрицей A. Как упоминалось ранее, хорошим примером такой матрицы является матрица достаточно большого размера n (например, $n \approx 10^5 \sim 10^6$), на каждой строке которой имеется лишь небольшое число m ненулевых элементов (например, $m \approx 10 \sim 100$).

Не каждый метод, пригодный для плотных матриц, является подходящим для решения разреженных СЛАУ большой размерности.

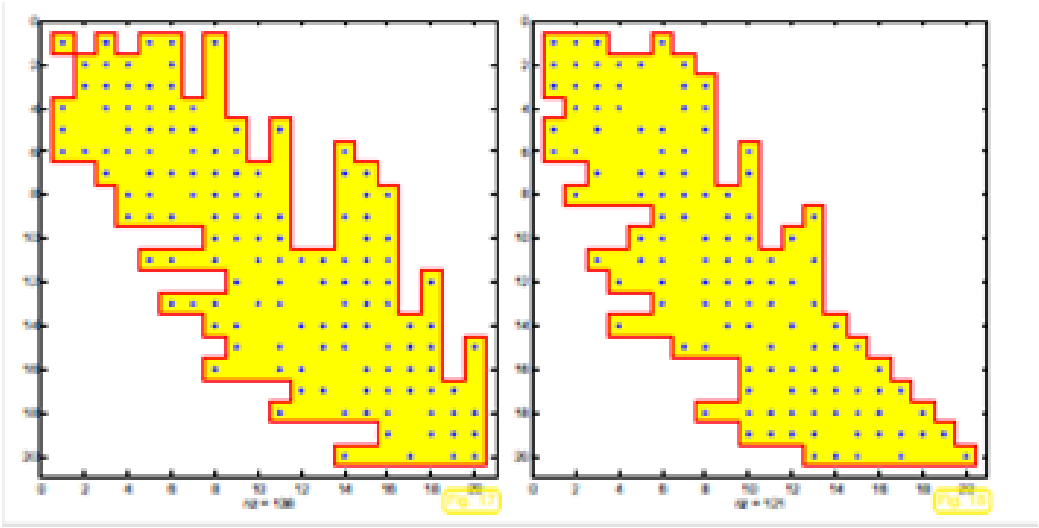

Рис. 1. Портреты и оболочки двух несимметричных матриц.

Интерес представляют только те методы, которые в процессе преобразований исходной матрицы сохраняют ее разреженность. Например, QR метод не является подходящим, т.к. после первых нескольких шагов матрица может стать плотной.

1. О заполнении при LU разложении. Будем предполагать, для упрощения изложения, что существует разложение $A = LU$.

Множество индексов

$$P(A) = \{(i,j) : \ a_{ij} \neq 0, \ i,j = 1:n\}$$

называется портретом матрицы A. Располагая в позициях $P(A)$ матрицы размера $n \times n$ некоторый символ (например, точку), получим графическое изображение ее портрета. На рис. 1 указаны портреты двух несимметричных матриц.

Если $a_{ij} = 0$, но соотвествующий элемент $l_{ij} \neq 0$ или $u_{ij} \neq 0$, то говорят, что в позиции (i,j) произошло заполнение при LU-разложении. Практически важный вопрос заключается в том, насколько велико заполнение и можно ли предсказать заполнение или его отсутствие в определенных позициях?

Для ответа на последний вопрос, определим $\ell_i(A) = \min\{j \leqslant i : \ a_{ij} \neq 0\}$ — номер столбца первого ненулевого элемента на i-той строке. Аналогично определим $m_j(A) = \min\{j \geqslant i : \ a_{ij} \neq 0\}$ — номер строки первого ненулевого элемента в j-том столбце. Следующее множество индексов назовем оболочкой A:

$$E(A) = \{(i,j) : \ell_i(A) \leqslant j \leqslant i = 1:n\} \cup \{(i,j) : m_j(A) \leqslant i \leqslant j = 1:n\}.$$

Элементы A с индексами из оболочки могут быть как ненулевыми, так и нулевыми. Важно, что если $(i,j) \notin E(A)$, то обязательно $a_{ij} = 0$. На рис. 1 цветом выделены оболочки матриц.

Теорема 7. *$E(A) = E(L+U)$, т.е. оболочки матриц A и $L+U$ совпадают. Как следствие заполнение при треугольном разложении может происходить только в позициях из оболочки A.*

Доказательство. Доказательство этой простой, но практически важной теоремы непосредственно следует из рассмотрения расчетных формул LU разложения и вынесено в упражнения. □

Следствие 1. Пусть A — трехдиагональная матрица, т.е. $a_{ij} = 0$, если $|i-j| > 1$ (т.е. ненулевые элементы A могут располагаться только на главной диагонали и двух соседних с ней диагоналях) и $A = LU$. Тогда ненулевые элементы L расположены только на главной диагонали и первой поддиагонали; ненулевые элементы U — только на главной диагонали и первой наддиагонали.

2. Предсказание заполнения. Разреженные матрицы A, L и U хранятся в ЭВМ в специальном формате. Чтобы распределить память под хранение L и U необходимо уметь оценивать заполнение. Для этого используется следующая гипотеза: в формулах типа

$$l_{ij} = \Big(a_{ij} - \sum_{k=1}^{j-1} l_{ik}u_{kj}\Big)/u_{jj}, \tag{41}$$

ненулевые слагаемые взаимно не уничтожаются (т.е., если a_{ij} или хотя бы одно произведение $l_{ik}u_{kj}$ в сумме $\neq 0$, то считается, что $l_{ij} \neq 0$). Это хорошее предположение, т.к. взаимное уничтожение плавающих чисел маловероятно. На основе этой гипотезы получается оценка заполнения. Важно, что при этом анализе можно оперировать только с портретом матрицы, т.е. все операции — целочисленные.

3. О перестановках. Чем меньше заполнение, которое зависит от портрета матрицы, тем эффективнее LU-разложение. В связи с этим рассмотрим СЛАУ $Ax = b$. Если переставить в каком-либо порядке ее строки, то получим СЛАУ с другой матрицей, но с тем же решением x. Ясно, что портреты матриц изменились, изменилось и заполнение. Переставим также столбцы матрицы. Получим систему $B\tilde{x} = \tilde{b}$, причем x и $\tilde{x}$ совпадают с точностью до соотвествующих перестановок компонент. Мы можем выбирать, какую систему решать; очевидно, надо выбрать ту, которая приведет к меньшему заполнению в множителях.

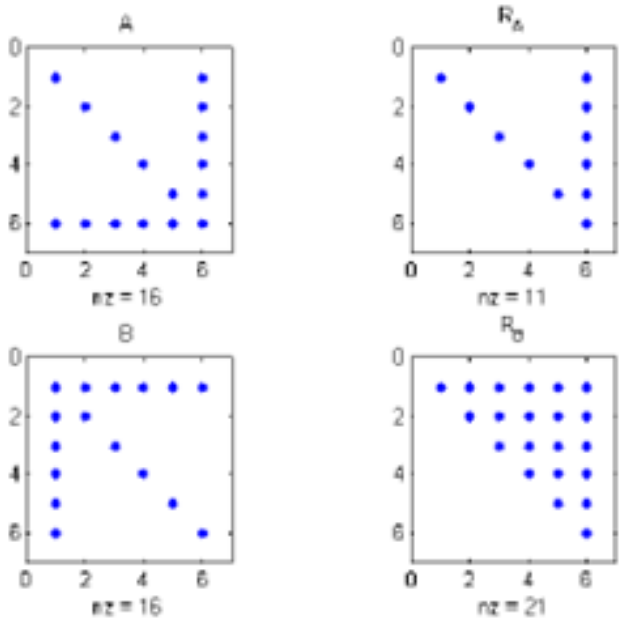

Рис. 2. Матрица с симметричным портретом (слева) и портрет ее множителя U (справа) до перестановки (верхние матрицы) и после (нижние матрицы).

На рисунке 2 сверху приведена матрица A и множитель разложения $R_A = U$. Видно, что заполнения в U не произошло. В то же время, если поменять местами первую и последнюю строки, а так же первый и последний столбцы (для сохранения симметрии при разложении Холецкого), то множитель разложения окажется полностью заполненным (на рис. 2 снизу приведена матрица B и множитель разложения $R_B = U$).

Таким образом, ключевой вопрос заключается в поиске эффективных перестановок строк (и возможно, также столбцов), которые позволят минимизировать заполнение. Как правило, поиск таких эффективных перестановок приводит к NP-полным задачам, поэтому для их поиска применяются эвристические алгоритмы. Они не дают гарантированного оптимального решения, но позволяют существенно уменьшить заполнение в процессе разложения, а значит, сэкономить время и память.

4. Об этапах решения разреженных систем. Разреженные матрицы хранятся в памяти ЭВМ в специальном формате: хранятся ненулевые элементы в виде одномерного массива и информация об их индексах. Современные методы решения СЛАУ включают два этапа.

1) На первом этапе ищутся перестановки строк (и возможно, столбцов), которые позволят уменьшить заполнение в множителях

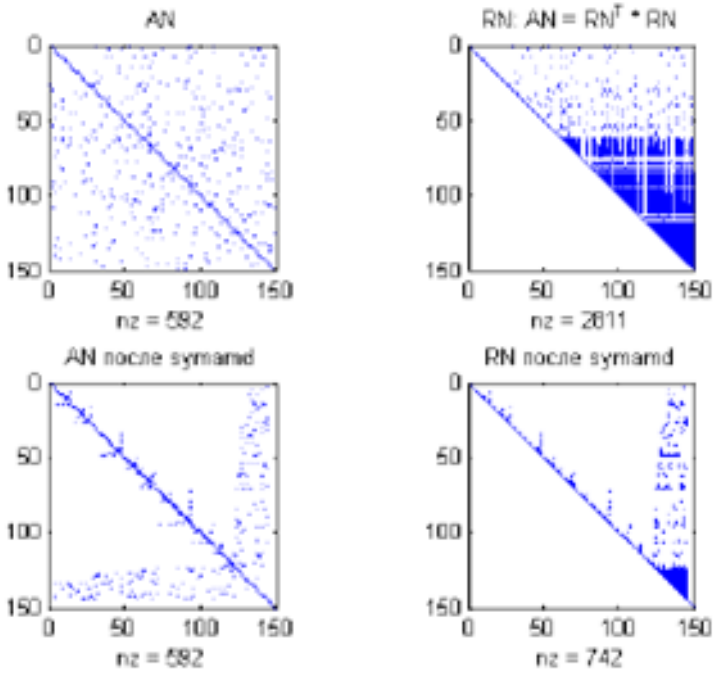

Рис. 3. Демонстрация эффективности алгоритма перенумерации symamd для симметричной матрицы. Наверху: исходная матрица и ее множитель Холецкого (трехкратное заполнение). Внизу матрица после перестановок и ее множитель (незначительное заполнение); nz — число ненулевых элементов матрицы.

разложения. Этот шаг целочисленный, операции производятся только с индексами. На этом шаге выделяется память и формируется портрет множителей (см. рис. (3), демонстрирующий эффективность MatLab-функции перестановок symamd).

2) На втором этапе вычисления производятся в плавающей арифметике и непосредственно вычисляются множители по формулам типа (41) но реализованным с учетом формата хранения матриц.

§ 9. О технологии разреженных матриц

Связанные с разреженными матрицами вопросы типа:

1) форматы хранения разреженных матриц;

2) эффективная реализация базовых операций линейно алгебры над матрицами в этих форматах и векторами (транспонирование, сумма, произведение матриц, произведение матрицы на вектор и т.д.);

3) решением задач линейной (матричной) алгебры (решение СЛАУ, задач на собственные значения и т.д.)

принято относить к технологии разреженных матриц. Кратко рассмотрим форматы хранения разреженных матриц.

1. Координатный формат. Наиболее очевидным способом хранения произвольной разреженной матрицы является координатный формат: хранятся только ненулевые элементы матрицы, и их координаты (номера строк и столбцов). В этом случае требуются три одномерных массива для хранения матрицы A:

1) массив ненулевых элементов матрицы A (обозначим его как a);

2) массив номеров строк матрицы A, соответствующих элементам массива a (обозначим его как i);

3) массив номеров столбцов матрицы A, соответствующих элементам массива a (обозначим его как j).

В качестве примера рассмотрим матрицу

```
A=[1  3  0  1
   0  4  0  2
   2  0  5  0  ];
```

которая может быть представлена в координатном формате как

```
(i,j)  a
(1,1)  1
(3,1)  2
(1,2)  3
(2,2)  4
(3,3)  5
(1,4)  1
(2,4)  2
```

Данный способ представления называют полным, поскольку представлена вся матрица A, и упорядоченным, поскольку ненулевые элементы матрицы перечислены по порядку по столбцам. Через $nnz(A)$ обозначим число ненулевых элементов матрицы A. В данном примере $nnz(A) = 7$.

2. Разреженный строчный формат. Это одна из наиболее широко используемых схем хранения разреженных матриц. Она предъявляет минимальные требования к памяти и в то же время оказывается очень удобной для нескольких важных операций над разреженными матрицами: сложения, умножения, перестановок строк, транспонирования, решения СЛАУ с разреженными матрицами коэффициентов как прямыми, так и итерационными методами и т. д.

В данном формате для хранения матрицы A требуется три одномерных массива:

1) массив ненулевых элементов матрицы A, в котором они перечислены подряд по строкам от первой до последней (обозначим его опять как a);

2) массив номеров столбцов для соответствующих элементов массива a (обозначим его как j);

3) массив указателей позиций в j, с которых начинается описание очередной строки (обозначим его p).

Таким образом, упорядоченные по возрастанию столбцевые индексы ненулевых элементов k-ой строки хранятся в векторе $\ell = p(k) : p(k+1) - 1$, а их значения в векторе $a(\ell) = [a(p(k)), a(p(k)+1), \ldots, a(p(k+1)-1)]$ (в обозначениях MatLab). Если матрица A состоит из n строк, то длина вектора p будет $n+1$, причем $p(n+1) = nnz(A)+1$. Данный способ представления также является полным и упорядоченным, поскольку элементы каждой строки хранятся в соответствии с возрастанием столбцевых индексов. Для нашего примера

```
p = [ 1 4 6 8 ]    j = [ 1 2 4 2 4 1 3 ]    a = [ 1 3 1 4 2 2 5 ]
```

Этот разреженный строчный формат (Compressed Sparse Row или CSR формат) обеспечивает эффективный доступ к строкам матрицы; доступ к столбцам по прежнему затруднен. Поэтому предпочтительно использовать этот способ хранения в тех методах, в которых преобладают строчные операции.

Иногда бывает удобно использовать полный неупорядоченный способ хранения, при котором внутри каждой строки элементы могут храниться в произвольном порядке. Результаты многих матричных операций получаются неупорядоченными, и упорядочивание может быть весьма затратным. В то же время, многие алгоритмы для разреженных матриц не требуют, чтобы представление было упорядоченным.

3. Разреженный столбцевой формат. В этом случае ненулевые элементы матрицы A перечисляются в порядке их появления в столбцах матрицы, а не в строках. Все ненулевые элементы хранятся по столбцам в массиве a; индексы строк ненулевых элементов – в массиве i; элементы массива p указывают на позиции, с которых начинается описание очередного столбца. Для нашего примера

```
i = [ 1 3 1 2 3 1 2 ]    p = [ 1 3 5 6 8 ]    a = [ 1 2 3 4 5 1 2 ]
```

Столбцевые представления могут рассматриваться как строчные представления транспонированных матриц. Разреженный столбцевой формат (CSC формат) обеспечивает эффективный доступ к столбцам матрицы; доступ к строкам затруднен. Поэтому предпочтительно использовать этот способ хранения в тех алгоритмах, в которых преобладают операции над столбцами матрицы

4. Умножение разреженной матрицы на плотный вектор. Пусть известна строчная форма A. Тогда следующая функция реализует требуемую операцию.

```
function y=Ax(p,j,a,x)
%% Ax : Умножение разреженной матрицы A на вектор x.
% p,j,a = строчная форма A в формате CSR

n = numel(p)-1;
y = zeros(n,1);
for k = 1:n
  for l=p(k):p(k+1)-1
    y(k) = y(k) + a(l)*x(j(l));
  end
end
```

§ 10. Метод прогонки

Рассмотрим СЛАУ с трехдиагональной матрицей A. Рассмотрим метод его решения, называемый методом прогонки. Произвольную систему с такой матрицей можно записать в следующем виде:

$$\begin{aligned} b_1x_1 + c_1x_2 &= f_1,\\ a_2x_1 + b_2x_2 + c_2x_3 &= f_2,\\ \dots\dots\dots\dots\dots\dots\dots\dots\\ a_ix_{i-1} + b_ix_i + c_ix_{i+1} &= f_i,\\ \dots\dots\dots\dots\dots\dots\dots\dots\\ a_nx_{n-1} + b_nx_n &= f_n. \end{aligned} \tag{42}$$

Разрешим первое уравнение системы относительно x_1. Получим:

$$x_1 = \alpha_2x_2 + \beta_2, \tag{43}$$

где

$$\alpha_2 = -\frac{c_1}{b_1}, \quad \beta_2 = \frac{f_1}{b_1}. \tag{44}$$

Используя соотношение (43) и второе уравнение системы (42), получим аналогичное выражение для x_2. Вообще, если $x_{i-1} = \alpha_i x_i + \beta_i$, то из i-го уравнения системы (42) получим

$$x_i = \alpha_{i+1} x_{i+1} + \beta_{i+1}, \quad i = 1 : n-1, \tag{45}$$

где

$$\alpha_{i+1} = -\frac{c_i}{b_i + a_i \alpha_i}, \quad \beta_{i+1} = \frac{f_i - a_i \beta_i}{b_i + a_i \alpha_i}, \quad i = 2 : n-1. \tag{46}$$

Используя (44) и (46), можно найти все $\alpha_i, \beta_i,\ i = 2 : n$. Записывая теперь соотношение (45) при $i = n-1$ и подставляя результат в последнее уравнение системы (42), получим

$$x_n = (a_n \beta_n - f_n)/(b_n - a_n \alpha_n).$$

Наконец, используя формулы (45) для $i = n-1, n-2, \ldots, 1$, найдем все остальные компоненты вектора x.

Рассмотренный метод есть вариант метода Гаусса, записанный применительно к случаю системы с трехдиагональной матрицей. Процесс вычислений α_i, β_i соответствует прямому ходу метода Гаусса, а вычисления по формулам (45) соответствуют обратному ходу метода Гаусса. Нетрудно подсчитать, что трудоемкость метода равна примерно $8n$ flop.

Метод может быть реализован, когда все знаменатели в формулах (44), (46) отличны от нуля. Учитывая связь метода прогонки с методом Гаусса, можно сказать, что данное условие выполнено, например, когда матрица системы (42) — матрица с диагональным преобладанием, т. е. $|c_1| < |b_1|$, $|a_n| < |b_n|$, $|a_i| + |c_i| < |b_i|$, $i = 2 : n-1$.

Задания для самостоятельной работы

ВОПРОСЫ ДЛЯ САМОКОНТРОЛЯ

1. Какие матрицы называются разреженными. Приведите примеры.
2. Дайте определение портрета разреженной матрицы.
3. Дайте определение оболочки разреженной матрицы.
4. Что понимается под заполнением множителей в процессе LU разложения матрицы?
5. Какая имеется связь LU разложения матрицы и ее оболочки?
6. Зависит ли заполнение множителей в процессе LU разложения от престановки строк матрицы? Приведите примеры.

7. Укажите портреты матриц L и U в LU-разложении трехдиагональной матрицы.

8. Опишите основные этапы решения разреженных систем уравнений.

9. Опишите координатный формат хранения разреженной матрицы. Для каких целей используется этот формат?

10. Опишите разреженный строчный формат хранения разреженной матрицы. Для решения каких задач выгоден этот формат?

11. Опишите разреженный столбцевой формат хранения разреженной матрицы. Для решения каких задач выгоден этот формат?

12. Для решения какой задачи применяется метод прогонки? Какова его трудоемкость?

ЗАДАЧИ И УПРАЖНЕНИЯ

1. Докажите теорему 7.

Указание. a) Рассмотрите компактную схему вычисления l_{ij} по формулам ijk-алгоритма: для всех $i = 1 : n$

$$l_{ij} = \Big(a_{ij} - \sum_{k=1}^{j-1} l_{ik}u_{kj}\Big)/u_{jj}, \quad j = 1 : i-1. \tag{47}$$

Последовательно полагая $j = 1 : i-1$, убедитесь, что $l_{ij} = 0$ для всех $j < \ell_i(A)$.
b) Аналогично, используя формулы jik-алгоритма: для всех $j = 1 : n$

$$u_{ij} = a_{ij} - \sum_{k=1}^{i-1} l_{ik}u_{kj}, \quad i = 1 : j, \tag{48}$$

убедитесь, что $u_{ij} = 0$, если $1 \leqslant i \leqslant m_j(A)$. Отсюда следует утверждение теоремы.

2. Матрицы с ненулевыми элементами на $2m+1$ диагоналях, прилегающих к главной (с индексами $|i-j| \leqslant m$), называются ленточными (с полушириной m).

a) Укажите на рисунке портреты таких матриц и портреты их треугольных множителей.

b) Как такие матрицы можно экономно хранить в ЭВМ?

c) Убедитесь, что трудоемкость их LU разложения равна $O(nm^2)$ flop.

§ 11. Нормы векторов и матриц

Говорят, что на пространстве $\mathbb{R}^n$ введена *норма*, если каждому вектору $x \in \mathbb{R}^n$ однозначно поставлено в соответствие вещественное число $\|x\|$ (читается: норма x). При этом должны быть выполнены следующие условия (*аксиомы нормы*):

1) $\|x\| \geqslant 0$ для $\forall\, x \in \mathbb{R}^n$; равенства $\|x\| = 0$ и $x = 0$ эквивалентны;

2) $\|\alpha x\| = |\alpha|\,\|x\|$ для $\forall\, x \in \mathbb{R}^n$, $\alpha \in \mathbb{R}$;

3) $\|x + y\| \leqslant \|x\| + \|y\|$ для $\forall\, x, y \in \mathbb{R}^n$.

Условие 3) называют *неравенством треугольника*. Отметим, что

4) $\big|\|x\| - \|y\|\big| \leqslant \|x - y\| \quad \forall\, x, y \in \mathbb{R}^n$.

Это неравенство вытекает из аксиомы 3). В самом деле,

$$\|x\| = \|x - y + y\| \leqslant \|x - y\| + \|y\|.$$

Аналогично, $\|y\| \leqslant \|x - y\| + \|x\|$. Неравенство 4) есть просто более краткая запись этих неравенств.

1. Примеры норм на $\mathbb{R}^n$. 1) Пусть $p \geqslant 1$. Равенство $\|x\|_p = \Big(\sum_{k=1}^n |x_k|^p\Big)^{1/p}$ определяет норму. Действительно, аксиомы 1), 2) выполнены очевидным образом, а неравенство 3) при $p = 1$ непосредственно вытекает из свойств модуля, а при $p > 1$ совпадает с известным неравенством Минковского, доказательство которого приведено в конце лекции. Отметим, что случай $p = 2$ соответствует Евклидовой норме вектора, хорошо известной из курса линейной алгебры: $\|x\|_2^2 = |x|^2 = (x, x)$ для любого $x \in \mathbb{R}^n$. Здесь $(\cdot, \cdot)$ — стандартное скалярное произведение на пространстве $\mathbb{R}^n$.

2) Положим $\|x\|_\infty = \max\limits_{1 \leqslant k \leqslant n} |x_k|$. Легко проверяется, что это равенство определяет норму, причем $\|x\|_\infty = \lim_{p \to \infty} \|x\|_p$ (см. упр. 1).

3) Пусть T — произвольная невырожденная матрица, а $\|\cdot\|$ — заданная норма на $\mathbb{R}^n$. Положим $\|x\|_* = \|Tx\|$. Легко видеть, что это равенство определяет новую норму вектора (см. упр. 2).

Определение 2. *Нормы $\|\cdot\|_{(1)}$ и $\|\cdot\|_{(2)}$ эквивалентны, если найдутся положительные постоянные c_1 и c_2 такие, что*[1)]

$$c_1\|x\|_{(1)} \leqslant \|x\|_{(2)} \leqslant c_2\|x\|_{(1)} \quad \forall\, x \in \mathbb{R}^n.$$

Теорема 8. (без доказательства) *Любые две нормы на пространстве $\mathbb{R}^n$ эквивалентны.*

Приведем, например, следующие оценки (см. упр. 3):

$$\|x\|_\infty \leqslant \|x\|_p \leqslant n^{1/p}\,\|x\|_\infty \quad \forall\, x \in \mathbb{R}^n, \quad p \geqslant 1. \tag{49}$$

[1)] Важно иметь в виду, что постоянные c_1, c_2 могут зависеть от n, т. е. от размерности $\mathbb{R}^n$.

2. Норма, порожденная скалярным произведением. Говорят, что на пространстве $\mathbb{R}^n$ введено *скалярное произведение*, если каждой паре векторов $x, y \in \mathbb{R}^n$ однозначно поставлено в соответствие вещественное число (x, y) (читается: скалярное произведение x и y). При этом должны быть выполнены следующие *аксиомы*:

1) $(x, x) \geqslant 0$ для $\forall\, x \in \mathbb{R}^n$; $(x, x) = 0 \quad \rightleftharpoons x = 0$;

2) $(x, y) = (y, x)$ для $\forall\, x, y \in \mathbb{R}^n$;

3) $(\alpha x + \beta y, z) = \alpha(x, z) + \beta(y, z)$ для $\forall\, x, y, z \in \mathbb{R}^n$, $\forall\, \alpha, \beta \in \mathbb{R}$.

Из курса линейной алгебры хорошо известно, что каждое скалярное произведение порождает норму $\|\cdot\|$ по правилу $\|x\| = (x, x)^{1/2}$. Такая норма связана со скалярным произведением неравенством Коши–Буняковского: $|(x, y)| \leqslant \|x\|\, \|y\|$.

Хорошо известный пример скалярного произведения — эвклидово скалярное произведение: $(x, y) = \sum_{i=1}^{n} x_i y_i$. Неравенство Коши–Буняковского в этом случае имеет вид:

$$\Big| \sum_{i=1}^{n} x_i y_i \Big|^2 \leqslant \Big(\sum_{i=1}^{n} x_i^2 \Big) \Big(\sum_{i=1}^{n} y_i^2 \Big).$$

Другой важный пример скалярного произведения — *энергетическое скалярное произведение* $(x, y)_A$. Оно порождается симметричной положительно определенной матрицей A по правилу (см. упр. 4)

$$(x, y)_A = (Ax, y) = \sum_{i,j=1}^{n} a_{ij} x_j y_i.$$

Порождаемая ею норма $\|x\|_A = (x, x)_A^{1/2}$ называется *энергетической нормой* вектора. Согласно неравенству Коши–Буняковского справедлива оценка: $|(x, y)_A| \leqslant \|x\|_A\, \|y\|_A$.

3. Важные числовые неравенства. Напомним, что функция $f(x)$ называется выпуклой на интервале (a, b), если для любых x_1, $x_2 \in (a, b)$ и для $\forall\, \lambda \in [0, 1]$ выполнено неравенство

$$f(\lambda x_1 + (1 - \lambda) x_2) \leqslant \lambda f(x_1) + (1 - \lambda) f(x_2).$$

Геометрически это означает, что любая точка графика функции f на отрезке $[x_1, x_2]$ лежит ниже хорды, стягивающей точки $(x_1, f(x_1))$, $(x_2, f(x_2))$, или на этой же хорде. Например, если функция f непрерывна и дважды непрерывно дифференцируема на (a, b), а ее вторая производная неотрицательна, тогда f — выпуклая функция на (a, b).

Теорема 9. *1) Пусть $a, b > 0$, $p > 1$, $1/p + 1/q = 1$, тогда*

$$ab \leqslant a^p/p + b^q/q \quad \textit{(неравенство Юнга).} \tag{50}$$

2) Неравенство Гельдера. Пусть $a, b \in \mathbb{R}^n$, $p > 1$, $1/p + 1/q = 1$. Тогда

$$\sum_{i=1}^{n} \left|a_i b_i\right| \leqslant \Big(\sum_{i=1}^{n} |a_i|^p\Big)^{1/p} \Big(\sum_{i=1}^{n} |b_i|^q\Big)^{1/q}. \tag{51}$$

3) Неравенство Минковского. Пусть $a, b \in \mathbb{R}^n$, $p > 1$. Тогда

$$\Big(\sum_{i=1}^{n} |a_i + b_i|^p\Big)^{1/p} \leqslant \Big(\sum_{i=1}^{n} |a_i|^p\Big)^{1/p} + \Big(\sum_{i=1}^{n} |b_i|^p\Big)^{1/p}. \tag{52}$$

Доказательство. 1) Легко видеть, что функция $-\ln(x)$ выпукла на интервале $(0, +\infty)$. Поэтому ($\lambda = 1/p$, $(1 - \lambda) = 1/q$))

$$\ln(a^p/p + b^q/q) \geqslant \ln(a^p)/p + \ln(b^q)/q = \ln(ab),$$

что равносильно (50). При $p = q = 2$ неравенство Гельдера называют также неравенством Коши–Буняковского.

2) Неравенство (51) выполнено, если хотя бы один из векторов a, b равен нулю. Иначе, используя неравенство Юнга, будем иметь:

$$\frac{|a_i|}{\Big(\sum_{i=1}^n |a_i|^p\Big)^{1/p}} \; \frac{|b_i|}{\Big(\sum_{i=1}^n |b_i|^q\Big)^{1/q}} \leqslant \frac{|a_i|^p}{p\sum_{i=1}^n |a_i|^p} + \frac{|b_i|^q}{q\sum_{i=1}^n |b_i|^q}\,. \tag{53}$$

Суммируя все эти неравенства, получим искомую оценку.

3) Будем считать a, b такими, что левая часть неравенства (52) положительна, так как в противном случае неравенство (52) выполняется очевидным образом. Ясно, что

$$\sum_{i=1}^{n} |a_i + b_i|^p = \sum_{i=1}^{n} |a_i + b_i|^{p-1}|a_i + b_i| \leqslant \sum_{i=1}^{n} |a_i + b_i|^{p-1}|a_i| + \sum_{i=1}^{n} |a_i + b_i|^{p-1}|b_i|\,. \tag{54}$$

Оценим правую часть, используя неравенство Гёльдера. Имеем

$$\sum_{i=1}^{n} |a_i + b_i|^{p-1}|a_i| \leqslant \Big(\sum_{i=1}^{n} |a_i + b_i|^{(p-1)q}\Big)^{1/q} \Big(\sum_{i=1}^{n} |a_i|^p\Big)^{1/p}.$$

Учтем здесь, что $(p-1)q = p$. Аналогично оценим второе слагаемое в правой части (54). В результате получим:

$$\sum_{i=1}^{n} |a_i + b_i|^p \leqslant \Big(\sum_{i=1}^{n} |a_i + b_i|^p\Big)^{1/q} \left(\Big(\sum_{i=1}^{n} |a_i|^p\Big)^{1/p} + \Big(\sum_{i=1}^{n} |b_i|^p\Big)^{1/p}\right).$$

Отсюда следует (52), т.к. $1 - 1/q = 1/p$. $\square$

4. Нормы матриц. Обозначим через M_n множество всех матриц размера $n \times n$. Определяя на нем обычным образом операции сложения двух матриц и умножения матрицы на число, превратим M_n в линейное пространство размерности n^2. Введем на нем норму, т. е. поставим в соответствие каждой $A \in M_n$ число $\|A\|$ (матричную норму) так, что для любых матриц $A, B \in M_n$ и чисел $\alpha \in \mathbb{R}$:

1) $\|A\| \geqslant 0$, равенства $\|A\| = 0$ и $A = 0$ эквивалентны;

2) $\|\alpha A\| = |\alpha| \|A\|$;

3) $\|A + B\| \leqslant \|A\| + \|B\|$;

4) $\|AB\| \leqslant \|A\| \|B\|$.

Замечание 4. Если выполнены только аксиомы 1-3, то говорят, что на M_n введена *векторная норма*. Не всякая векторная норма является матричной. Пусть, например,

$$\|A\| = \max_{1 \leqslant i,j \leqslant n} |a_{ij}|. \tag{55}$$

Очевидно, это — векторная норма, но она не является матричной, т.к., если

$$A = \begin{bmatrix} 1 & 1 \\ 1 & 1 \end{bmatrix}, \text{ то } AA = \begin{bmatrix} 2 & 2 \\ 2 & 2 \end{bmatrix},$$

$\|A\| = 1$, $\|AA\| = 2$, и неравенство $\|AA\| \leqslant \|A\| \|A\|$ не выполнено.

5.1. Примеры матричных норм.

a) Положим $\|A\|_{l_1} = \sum\limits_{i,j=1}^{n} |a_{ij}|$. Очевидно, три первых аксиомы нормы выполнены. Проверим аксиому 4). По определению имеем

$$\|AB\|_{l_1} = \sum_{i,j=1}^{n} \left| \sum_{k=1}^{n} a_{ik} b_{kj} \right| \leqslant \sum_{i,j,k=1}^{n} |a_{ik}||b_{kj}| \leqslant$$

$$\leqslant \sum_{i,j,k,m=1}^{n} |a_{ik}||b_{mj}| = \|A\|_{l_1} \|B\|_{l_1}.$$

b) Положим $\|A\|_E = \left(\sum_{i,j=1}^{n} |a_{ij}|^2 \right)^{1/2}$. Эта норма порождается естественным скалярным произведением на пространстве $\mathbb{R}^{n^2}$, поэтому три первых аксиомы для нее выполняются. Норму $\|A\|_E$ обычно называют *евклидовой* нормой или нормой *Фробениуса*[1)]. Проверим

[1)] Фердинанд Георг Фробениус (Ferdinand Georg Frobenius; 1849 — 1917) — немецкий математик.

аксиому 4), опираясь на неравенство Коши-Буняковского. Имеем

$$\|AB\|_E^2 = \sum_{i,j=1}^{n} \Big| \sum_{k=1}^{n} a_{ik} b_{kj} \Big|^2 \leqslant \sum_{i,j=1}^{n} \sum_{k=1}^{n} |a_{ik}|^2 \sum_{k=1}^{n} |b_{kj}|^2 =$$
$$= \sum_{i,k=1}^{n} |a_{ik}|^2 \sum_{k,j=1}^{n} |b_{kj}|^2 = \|A\|_E^2 \|B\|_E^2.$$

c) Пусть задана норма $\|\cdot\|$ на $\mathbb{R}^n$. Матричную норму

$$\|A\| = \max_{x \in \mathbb{R}^n,\, x \neq 0} \frac{\|Ax\|}{\|x\|} = \max_{x \in \mathbb{R}^n,\, x \neq 0} \left\| A \frac{x}{\|x\|} \right\| = \max_{\|x\|=1} \|Ax\|, \qquad (56)$$

называют *подчиненной* норме векторов $\|\cdot\|$ или *операторной* нормой.

То, что максимум в (56) достигается, оставим без доказательства, а проверку аксиом 1-3) вынесем в упр. 3. Проверим аксиому 4). Первоначально заметим, что

$$\|A\| = \max_{x \neq 0} \frac{\|Ax\|}{\|x\|} \geqslant \frac{\|Ay\|}{\|x\|} \quad \forall\, y \in \mathbb{R}^n.$$

Отсюда следует важное свойство подчиненной нормы:

5) $\|Ax\| \leqslant \|A\|\,\|x\|$ для любых $x \in \mathbb{R}^n$.

Свойство 5) позволяет проверить аксиому 4). Имеем $\|ABx\| = \|A(Bx)\| \leqslant \|A\|\,\|Bx\| \leqslant \|A\|\,\|B\|\,\|x\|$|. Поэтому

$$\|AB\| = \max_{x \neq 0} \frac{\|ABx\|}{\|x\|} \leqslant \|A\|\,\|B\|.$$

Ясно, что при любом способе задания нормы на $\mathbb{R}^n$ подчиненная норма единичной матрицы равна единице.

Не всякая матричная норма подчинена какой либо норме векторов. Например, норма Фробениуса не подчинена никакой норме векторов, так как $\|I\|_E = \sqrt{n}$.

5.2. Примеры подчиненных матричных норм.

а) Пусть норма на пространстве $\mathbb{R}^n$ определена равенством $\|x\|_1 = \sum\limits_{k=1}^{n} |x_k|$. Тогда подчиненная норма матрицы есть

$$\|A\|_1 = \max_{x \in \mathbb{R}^n,\, \|x\|_1 = 1} \|Ax\|_1.$$

Нетрудно видеть, что для любого вектора $x \in \mathbb{R}^n$, $\|x\|_1 = 1$,

$$\|Ax\|_1 = \sum_{i=1}^{n} \Big| \sum_{j=1}^{n} a_{ij} x_j \Big| \leqslant \sum_{i=1}^{n} \sum_{j=1}^{n} |a_{ij}||x_j| = \sum_{j=1}^{n} |x_j| \sum_{i=1}^{n} |a_{ij}| \leqslant$$
$$\leqslant \sum_{j=1}^{n} |x_j| \max_{j=1:n} \sum_{i=1}^{n} |a_{ij}| = \max_{j=1:n} \sum_{i=1}^{n} |a_{ij}| = S.$$

Пусть $S = \sum\limits_{i=1}^{n} |a_{ik}|$ и $e_k = (0, \ldots, 0, 1, 0, \ldots, 0)^T$ есть орт k-той координатной оси. Ясно, что $\|e_k\|_1 = 1$, а $\|Ae_k\|_1 = \sum\limits_{i=1}^{n} |a_{ik}| = S$. Таким образом, $\|Ax\|_1 \leqslant S$ для всех x, $\|x\|_1 = 1$, и $\|Ae_k\|_1 = S$, $\|e_k\|_1 = 1$. Поэтому

$$\|A\|_1 = \max_{\|x\|_1 = 1} \|Ax\|_1 = \max_{j=1:n} \sum_{i=1}^{n} |a_{ij}|.$$

Эту норму $\|A\|_1$ часто называют *столбцовой* нормой матрицы A.

b) Определим норму на $\mathbb{R}^n$ равенством $\|x\|_\infty = \max\limits_{k=1:n} |x_k|$. Тогда для любого $x \in \mathbb{R}^n$ такого, что $\|x\|_\infty = 1$

$$\|Ax\|_\infty = \max_{i=1:n} \Big| \sum_{j=1}^{n} a_{ij} x_j \Big| \leqslant \max_{i=1:n} \sum_{j=1}^{n} |a_{ij}||x_j| \leqslant$$
$$\leqslant \max_{j=1:n} |x_j| \max_{i=1:n} \sum_{j=1}^{n} |a_{ij}| = \max_{i=1:n} \sum_{j=1}^{n} |a_{ij}| = S.$$

Итак, $\|Ax\|_\infty \leqslant S$ при любом x, $\|x\|_\infty = 1$. Докажем, что найдется e такой, что $\|e\|_\infty = 1$ и $\|Ae\|_\infty = S$. Тогда получим, что $\|A\|_\infty = S$.

Пусть $S = \sum\limits_{j=1}^{n} |a_{kj}|$. Определим компоненты e как

$$e_j = \begin{cases} a_{kj}/|a_{kj}|, & a_{kj} \neq 0, \\ 1, & a_{kj} = 0, \end{cases} \qquad j = 1 : n.$$

Ясно, что $\|e\|_\infty = 1$, причем что для любого $i = 1 : n$

$$\Big| \sum_{j=1}^{n} a_{ij} e_j \Big| \leqslant \sum_{j=1}^{n} |a_{ij}| \leqslant \max_{i=1:n} \sum_{j=1}^{n} |a_{ij}| = S, \tag{57}$$

а для $i = k$ по определению e_j получим

$$\Big|\sum_{j=1}^{n} a_{ij}e_j\Big| = \sum_{j=1}^{n} |a_{kj}| = S. \tag{58}$$

Из (57), (58) следует $\|Ae\|_\infty = S$. Таким образом,

$$\|A\|_\infty = \max_{\|x\|_\infty=1} \|Ax\|_\infty = \max_{i=1:n} \sum_{j=1}^{n} |a_{ij}|.$$

Норму $\|A\|_\infty$ часто называют *строчной* нормой матрицы A.

c) Введем матричную норму, подчиненную евклидовой норме вектора $\|x\|_2 = (x,x)^{1/2}$. Для любого $x \in \mathbb{R}^n$, $\|x\|_2 = 1$, справедливо равенство $\|Ax\|_2^2 = (Ax, Ax) = (A^TAx, x)$. Матрица $S = A^TA$ симметрична и положительно определена. Согласно спектральному разложению $S = H^T\Lambda H$, где H — ортогональная матрица, $\Lambda = \mathrm{diag}(\lambda_1, \ldots, \lambda_n)$ диагональная матрица из собственных чисел S. Поэтому

$$\|Ax\|_2^2 = (H^T\Lambda Hx, x) = (\Lambda Hx, Hx) = (\Lambda y, y) = \sum_{i=1}^{n} \lambda_i y_i^2 \leqslant \max_{i=1:n} \lambda_i, \tag{59}$$

т.к. $y = Hx$, $\|y\|_2 = \|x\|_2 = 1$. С другой стороны, если $\max_{i=1:n} \lambda_i = \lambda_k$, то выбирая в (59) x как решение уравнения $Hx = e_k$, где e_k есть орт k-той оси, получим $\|Ax\|_2^2 = \lambda_k$, что вместе с (59) приводит к равенству

$$\|A\|_2 = \max_{\|x\|_1=1} \|Ax\|_2 = \sqrt{\max_{i=1:n} \lambda_i(A^TA)},$$

где $\lambda_i(A^TA)$ есть собственное число матрицы A^TA.

Отметим следующий важный для многих приложений частный случай симметричной матрицы, когда $A = A^T$. В этом случае, из равенства $Ax = \lambda(A)x$ вытекает $A^2x = \lambda(A)Ax = \lambda^2(A)x$. Поэтому

$$A = A^T \quad \Rightarrow \quad \|A\|_2 = \max_{\|x\|_1=1} \|Ax\|_2 = \max_{i=1:n} |\lambda_i(A)|,$$

где через $\lambda_i(A)$ обозначены собственные числа матрицы A.

Максимальное по модулю собственное число матрицы A принято обозначать через $\rho(A)$ и называть спектральным радиусом матрицы A. Норму $\|A\|_2$ в связи с этим часто называют *спектральной*.

Замечание 5. Вычисление собственных чисел матрицы, вообще говоря, — довольно сложная задача. Поэтому полезно получить некоторую оценку величины $\|A\|_2$, просто выражаемую через элементы матрицы A.

Теорема 10. *Для любой матрицы A справедливо неравенство $\|A\|_2 \leqslant \|A\|_E$.*

Доказательство. Используем стандартное обозначение $\mathrm{tr}(S)$ для следа матрицы S, вычисляемого как сумма элементов ее главной диагонали. Известно, что он равен сумме собственных чисел S. Поэтому $\mathrm{tr}(A^TA) = \sum_{k=1}^{n} \lambda_k(A^TA) \geqslant \max_{k=1:n} \lambda_k(A^TA) = \|A\|_2^2$. С другой стороны легко вычислить, что $\mathrm{tr}(A^TA) = \sum\limits_{i,j=1}^{n} |a_{ij}|^2$. Следовательно,

$$\|A\|_2 \leqslant \Big(\sum_{i,j=1}^{n} |a_{ij}|^2 \Big)^{1/2} = \|A\|_E. \quad \square$$

§ 12. Устойчивость решений СЛАУ

Когда мы говорим о решении СЛАУ $Ax = b$ в практическом смысле, то должны отдавать себе отчет в том, что данные задачи, а именно, матрица A и вектор правой части b, заданы приближенно. То есть, вместо матрицы A известна матрица $\bar{A} = A + \Delta A$, а вместо вектора b задан вектор $\bar{b} = b + \Delta b$. Матрицу ΔA называют возмущением матрицы A, вектор Δb — возмущением вектора b. Причинами появления возмущений могут служить:

1) ввод чисел в ЭВМ. При этом возмущения имеют относительный порядок $\approx 10^{-16}$ (тип double). Это возмущение всегда присутствует, поскольку мы решаем задачу при помощи ЭВМ;

2) погрешности алгоритмов (если элементы матрицы A и вектора b вычисляются приближенно при помощи некоторого алгоритма). Величины возмущений зависят от точности используемых алгоритмов;

3) погрешности измерений (погрешности приборов, если элементы A и b получаются в результате измерений). Величина возмущений зависит от точности измерений, и т.д.

Таким образом, вместо системы $Ax = b$, на самом деле, мы решаем СЛАУ $\bar{A}\bar{x} = \bar{b}$. Решение этой возмущенной системы $\bar{x}$, конечно, не совпадает с решением исходной системы. Возникает естественный вопрос, на который мы получим некоторый ответ далее:

Если возмущения малы, т.е. данные задач $Ax = b$ и $\bar{A}\bar{x} = \bar{b}$ мало отличаются друг от друга, то будут ли близки их решения?

При ответе на этот вопрос мы будем предполагать, что обе задачи решаются точно, т.е. игнорируются ошибки методов их решения и ошибки округления при выполнении арифметических операций.

Теорема 11. *1) Пусть x и $\bar{x}$ есть решения систем $Ax = b$ и $A\bar{x} = \bar{b}$ соответственно,* $\det(A) \neq 0$. *Тогда*

$$\frac{\|x - \bar{x}\|}{\|x\|} \leqslant \operatorname{cond}(A) \frac{\|b - \bar{b}\|}{\|b\|}.$$

2) Если же $\bar{A}\bar{x} = b$, то

$$\frac{\|x - \bar{x}\|}{\|\bar{x}\|} \leqslant \operatorname{cond}(A) \frac{\|A - \bar{A}\|}{\|A\|}.$$

Здесь число $\operatorname{cond}(A) = \|A^{-1}\| \, \|A\|$ *называется числом обусловленности матрицы A, норма вектора — произвольная, норма матрицы — подчиненная норме вектора (операторная).*

Доказательство. 1) Имеем $A(x - \bar{x}) = b - \bar{b}$, Следовательно,

$$\|x - \bar{x}\| = \|A^{-1}(b - \bar{b})\| \leqslant \|A^{-1}\| \, \|b - \bar{b}\|. \tag{60}$$

С другой стороны, $\|b\| = \|Ax\| \leqslant \|A\| \, \|x\|$. Умножим это неравенство на (60) и поделим обе части полученного равенства на $\|x\| \, \|b\|$. Получим 1). Для доказательства 2) заметим, что

$$\bar{x} - x = (\bar{A}^{-1} - A^{-1})b = A^{-1}(A - \bar{A})\bar{A}^{-1}b = A^{-1}(A - \bar{A})\bar{x}.$$

Отсюда следует оценка

$$\|\bar{x} - x\| \leqslant \|A^{-1}\| \, \|(A - \bar{A})\| \, \|\bar{x}\|.$$

Поделим обе части этой оценки на $\|\bar{x}\|$ и умножим и поделим правую часть на $\|A\|$. Получим 2). $\square$

В общем случае справедлива аналогичная оценка

Теорема 12. *Пусть x и $\bar{x}$ есть решения систем $Ax = b$ и $\bar{A}\bar{x} = \bar{b}$, соответственно,* $\det(A) \neq 0$, $\triangle A = A - \bar{A}$, $\triangle b = b - \bar{b}$. *Тогда, если* $\|A^{-1}\triangle A\| < 1$, *то*

$$\frac{\|x - \bar{x}\|}{\|x\|} \leqslant \frac{\operatorname{cond}(A)}{1 - \|A^{-1}\triangle A\|} \Big(\frac{\|\triangle b\|}{\|b\|} + \frac{\|\triangle A\|}{\|A\|} \Big).$$

Как видим, оценка возмущения решения прямо пропорционально числу обусловленности матрицы, причем $cond(A) \geqslant 1$. В самом деле, $I = A\,A^{-1}$, т.е. $1 = \|I\| = \|A\,A^{-1}\| \leqslant \|A\|\,\|A^{-1}\| = \mathrm{cond}(A)$. У некоторых матриц $cond(A)$ может быть велико, в этом случае оценка не гарантирует малость возмущения решения. Конечно, мы получили лишь оценки, но на классе всех СЛАУ они являются точными.

Матрицы с очень большим числом обусловленности принято называть *плохо обусловленными*. СЛАУ с плохообусловленными матрицами требуют особого подхода и, как правило, не могут быть удовлетворительно решены рассмотренными нами прямыми методами.

Пример 1. Матрица Гильберта — хороший пример плохо обусловленной матрицы.

$$H_n = \left\{\frac{1}{i+j-1}\right\}_{i,j=1}^{n}. \quad \text{СЛАУ} \quad Ax = b, \quad x = (1,1,\ldots,1)^T.$$

$$\det(H_n) \approx 0.6\,n^{-1/4}\,(2\pi)^n 4^{-n^2}, \quad \mathrm{cond}_2(H_n) = O(2.2^{4n}/\sqrt{n}).$$

n	4	8	10	12	15
cond_2	1.6e+5	1.5e+10	1.60e+13	1.7e+016	2.5e+17
err_2	1.9e-13	1.0e-7	2.7e-4	0.08	1.3

Таблица 1.1. Матрица Гильберта: $\mathrm{cond}_2 = \mathrm{cond}_2(H_n)$ и относительная погрешность решения СЛАУ err_2 в евклидовой норме.

В таб. 1.1 представлены результаты вычисления в MatLab числа обусловленности матрицы Гильберта, а также относительной погрешности решения СЛАУ: ее точное решение $x = (1,1,\ldots,1)^T$, $\bar{x} = H\backslash b$ — решение этой системы, вычисленное в MatLab. Отношение $err_2/\mathrm{cond}_2 \approx 10^{-17}$.

Пример 2. Рассмотрим СЛАУ $Ax = b$, где A — модифицированная Жорданова клетка ($a > 1$):

$$A = \begin{pmatrix} 1 & a & & & \\ & 1 & a & & \\ & & \ddots & & \\ & & & 1 & a \\ & & & & 1 \end{pmatrix}, \qquad A^{-1} = \begin{pmatrix} 1 & -a & a^2 & \cdots & (-1)^{n-1}a^{n-1} \\ & 1 & -a & a^2 & \\ & & \ddots & & a^2 \\ & & & 1 & -a \\ & & & & 1 \end{pmatrix}.$$

У обратной матрицы главная диагональ состоит из 1, наддиагонали соответственно состоят из $-a$, a^2, $-a^3$ и т.д.

Будем считать, что $b = (1+a, 1+a, \ldots, 1+a, a)^T$, так что известно точное решение СЛАУ $x = (1,1,\ldots,1)^T$, $\|x\|_\infty = 1$. Имеем,

$$\|A\|_\infty = 1 + a, \quad \|A^{-1}\|_\infty = 1 + a + a^2 + \ldots + a^{n-1} = \frac{a^n - 1}{a - 1},$$

$$\mathrm{cond}_\infty(A) = (1+a)\,\frac{a^n - 1}{a - 1}.$$

Рассмотрим систему $A\bar{x} = \bar{b}$, где $\bar{b} = b - \varepsilon e_n$, $e_n = (0, 0, \ldots, 0, 1)^T$.
Так как $\|b\|_\infty = 1 + a$, $\|b - \bar{b}\|_\infty = \varepsilon$, то согласно теореме 2 имеем

$$\frac{\|x - \bar{x}\|_\infty}{\|x\|_\infty} \leqslant \mathrm{cond}_\infty(A)\, \frac{\|b - \bar{b}\|_\infty}{\|b\|_\infty} = \frac{a^n - 1}{a - 1}\, \varepsilon \approx a^{n-1}\varepsilon. \tag{61}$$

С другой стороны, пусть $z = x - \bar{x}$. Тогда $Az = \varepsilon\, e_n$. Эта система легко решается обратным ходом. Получаем, $z_n = \varepsilon$, $z_{n-1} = -a\,\varepsilon$, $z_{n-2} = a^2\,\varepsilon$, …, $z_1 = \pm\, a^{n-1}\,\varepsilon$. Таким образом, $\|x - \bar{x}\|_\infty = \|z\|_\infty = a^{n-1}\,\varepsilon$ и

$$\frac{\|x - \bar{x}\|_\infty}{\|x\|_\infty} = a^{n-1}\,\varepsilon. \tag{62}$$

Как видим, теоретическая оценка (61) практически совпадает с точной величиной относительной погрешности (62).

Задания для самостоятельной работы

ВОПРОСЫ ДЛЯ САМОКОНТРОЛЯ

1. Дайте определение нормы векторов. Приведите примеры.

2. Дайте определение норм $\|x\|_1$, $\|x\|_2$, $\|x\|_\infty$.

3. Дайте определение эквивалентных норм векторов. Приведите примеры эквивалентных норм.

4. Дайте определение скалярного произведения векторов. Приведите примеры.

5. Дайте определение энергетического скалярного произведения.

6. Дайте определение энергетической нормы вектора. Какой матрицей оно порождается?

7. Что понимают под сходимостью последовательности векторов к некоторому вектору?

8. Что означает сходимость векторов в норме $\|x\|_\infty$?

9. Если последовательность векторов сходится к некоторому вектору в одной норме, сходится ли она в другой норме к тому же вектору?

10. Запишите неравенство Коши–Буняковского в общем случае. Приведите примеры.

11. Дайте определение векторной нормы матриц. Приведите примеры.

12. Дайте определение подчиненной нормы матрицы. Чему равна подчиненная норма единичной матрицы?

13. Дайте определение норм $\|A\|_1$, $\|A\|_2$, $\|A\|_\infty$.

14. Дайте определение нормы $\|A\|_2$ симметричной матрицы.

15. Что понимается под возмущение матрицы или вектора?

162. Дайте определение числа обусловленности матрицы.

17. Приведите оценку возмущения решения СЛАУ при возмущении ее правой части.

18. Приведите оценку возмущения решения СЛАУ при возмущении ее матрицы.

19. Какая матрица называется плохо обусловленной?

20. Приведите примеры плохо обусловленных матриц.

21. Связана ли плохая обусловленность матрицы с малостью ее определителя и как?

ЗАДАЧИ И УПРАЖНЕНИЯ

1. Докажите, что для любого $x \in \mathbb{R}^n$ выполнено предельное соотношение $\|x\|_\infty = \lim\limits_{p\to\infty} \|x\|_p$.

2. Проверьте аксиомы нормы $\|x\|_* = \|Tx\|$, $|T| \neq 0$.

3. Докажите справедливость оценок (49).

4. Докажите, что $(x, y)_A = (Ax, y)$ есть скалярное произведение, если A симметрична и положительно определена, а $(\cdot, \cdot)$ есть евклидово скалярное произведение.

5. Пусть $\|\cdot\|$ — матричная норма на M_n, $S \in M_n$ — произвольная невырожденная матрица. Докажите, что $\|A\|_{(s)} = \|SAS^{-1}\|$ для $\forall\, A \in M_n$ определяет матричную норму.

6. Доказать, что $\|A\| = n \max\limits_{1\leqslant i,j\leqslant n} |a_{ij}|$ является матричной (проверьте аксиомы 1-4).

7. Докажите, что для подчиненной нормы выполнены аксиомы 1-3 матричной нормы.

8. Найдите нормы $\|A\|_1$, $\|A\|_2$, $\|A\|_\infty$ матрицы

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

9. Найдите нормы $\|A\|_1$, $\|A\|_2$, $\|A\|_\infty$ матрицы

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

10. Докажите, что для любой матрицы A: 1) нормы $\|A\|_2$ и $\|A\|_E$ не меняются при умножении A (слева или справа) на любую ортогональную матрицу; 2) $\|A\|_2 = \|A^T\|_2$.

11. Постройте a) пример хорошо обусловленной матрицы 2×2. Дайте геометрическую интерпретацию решения СЛАУ с этой матрицей. b) Постройте плохо обусловленной матрицы 2×2. Дайте геометрическую интерпретацию решения СЛАУ с этой матрицей.

ГЛАВА 2

Итерационные методы решения систем уравнений

Основная идея итерационных методов (ИМ) решения системы уравнений $Ax = b$ состоит в построении последовательности векторов x^0, x^1,...,x^k,..., сходящейся к ее решению x. Пусть $\|\cdot\|$ какая-либо заданная норма вектора.

Определение 1. *Последовательность векторов $\{x^k\}_{k=0}^{\infty}$ называется сходящейся к вектору x, если $\|x - x^k\| \to 0$ при $k \to \infty$.*

Если выполнена оценка $\|x - x^{k+1}\| \leqslant q\,\|x - x^k\|$, $k \geqslant 0$, $q < 1$, то говорят, что ИМ линейно сходится или сходится со скоростью геометрической прогрессии с основанием q.

Отметим, что поскольку всякая норма на пространстве $\mathbb{R}^n$ эквивалентна норме $\|\cdot\|_\infty$, то из сходимости последовательности векторов по любой норме вытекает ее покомпонентная сходимость. То есть, если $x^k = (x_1^k, \ldots, x_n^k)^T \to x = (x_1, \ldots, x_n)^T$ при $k \to \infty$, то для каждого $i = 1 : n$ также $x_i^k \to x_i$.

За приближенное решение СЛАУ принимается вектор x^k при достаточно большом k. Критерием окончания ИМ обычно является либо достаточная близость двух соседних приближений, либо достаточная малость вектора невязки $r^k = Ax^k - b$, т.е. итерации заканчиваются при выполнении одного из условий (или комбинации обоих):

$$\|x^k - x^{k-1}\| \leqslant \varepsilon, \quad \|Ax^k - b\| \leqslant \varepsilon$$

Вектор x^k называют k-тым приближением к решению или k-той итерацией (решения), k — номером итерации. ИМ состоит в циклическом выполнении одной и той же группы операций, скажем $\mathcal{F}$ (iteration — повтор, повторение). *Двуслойные ИМ* имеет вид

$$x^{k+1} = \mathcal{F}(x^k), \quad k = 0, 1, \ldots$$

Каждое следующее приближение вычисляется лишь по предыдущему и для начала счета необходимо знать (задать) лишь одно начальное приближение x^0 к решению СЛАУ. ИМ вида

$$x^{k+1} = \mathcal{F}(x^{k-1}, x^k), \quad k = 1, 2, \ldots$$

называются *трехслойными ИМ* и требуют двух начальных приближений x^0 и x^1. ИМ конструируются так, чтобы начальные приближения можно было задавать произвольно. Возникает естественный вопрос: зачем нужны ИМ, если имеются прямые методы, позволяющие найти решение СЛАУ за конечное число операций? Можно привести по крайней мере две причины, когда ИМ будут полезны.

1) При реализации прямых методов важно, чтобы исходные и промежуточные данные располагались в оперативной (быстрой) памяти компьютера. Если порядок системы настолько велик, что оперативной памяти для реализации метода недостаточно или число операций метода недопустимо велико, то для таких систем предпочтительнее оказываются ИМ, поскольку для их реализации достаточно уметь вычислять лишь произведение матрицы СЛАУ на вектор. Часто это можно сделать не храня в памяти ЭВМ самой матрицы и достаточно экономично реализуется для разреженных матриц.

2) На практике часто встречается ситуация, когда достаточно знать не точное, а приближенное решение СЛАУ, причем может иметься неплохое начальное приближение к решению. В этом случае ИМ может оказаться предпочтительным, если достаточно небольшое число итераций позволит получить решение с необходимой точностью.

§ 1. Простейшие итерационные методы.

Всюду в дальнейшем через z^k будем обозначать вектор $x - x^k$, где x — решение системы $Ax = b$, т.е. *погрешность приближения* с номером k. Далее используем представление матрицы A в виде суммы

$$A = L + D + U, \quad D = \operatorname{diag}(a_{11}, a_{22}, \ldots, a_{nn}),$$

где L (U) — нижняя (верхняя) треугольная матрица, все элементы ниже (выше) главной диагонали которой совпадают с соответствующими элементами матрицы A.

1. Метод Якоби.[1)]. Будем считать, что все диагональные элементы матрицы A отличны от нуля. Запишем систему $Ax = b$ в виде

$$Dx = b - Lx - Ux \quad \rightleftarrows \quad x = D^{-1}\big(b - Lx - Ux\big). \tag{1}$$

Итерационный метод определим по формуле

$$x^{k+1} = D^{-1}\big(b - Lx^k - Ux^k\big), \quad k = 0, 1, \ldots, \tag{2}$$

[1)] Карл Густав Якоб Яко́би (Carl Gustav Jacob Jacobi; 1804 — 1851) — немецкий математик.

где начальное приближение $x^0 = (x_1^0, x_2^0, \ldots, x_n^0)^T$ — произвольно задано. Компоненты приближения x^{k+1} определяются по уже найденному вектору x^k при помощи соотношений:

$$x_i^{k+1} = \Big(b_i - \sum_{j=1}^{i-1} a_{ij}\, x_j^k - \sum_{j=i+1}^{n} a_{ij}\, x_j^k\Big)/a_{ii}, \quad i = 1:n. \tag{3}$$

Формулы (3) определяют итерационный метод решения системы $Ax = b$, называемый *методом Якоби*.

Укажем легко проверяемое достаточное условие сходимости этого метода. Напомним, что для матрицы A выполнено условие диагонального преобладания по строкам, если

$$q = \max_{1\leqslant i\leqslant n} \sum_{j=1,\ j\neq i}^{n} \frac{|a_{ij}|}{|a_{ii}|} < 1. \tag{4}$$

Теорема 1. *Для матриц с диагональным преобладанием по строкам метод Якоби линейно сходится при любом начальном приближении. Справедлива следующая оценка скорости сходимости:*

$$\|x - x^k\|_\infty \leqslant q^k\, \|x - x^0\|_\infty. \tag{5}$$

Доказательство. Вычитая из второго равенства (1) равенство (2), получим $z^{k+1} = -D^{-1}\big(Lz^k + Uz^k\big)$ или

$$z_i^{k+1} = -\Big(\sum_{j=1}^{i-1} a_{ij}\, z_j^k - \sum_{j=i+1}^{n} a_{ij}\, z_j^k\Big)/a_{ii}, \quad i = 1:n,$$

следовательно, для всех $i = 1:n$,

$$|z_i^{k+1}| \leqslant \sum_{j=1}^{i-1} \frac{|a_{ij}|}{|a_{ii}|}\, |z_j^k| + \sum_{j=i+1}^{n} \frac{|a_{ij}|}{|a_{ii}|}\, |z_j^k| \leqslant$$

$$\leqslant \left(\sum_{j=1}^{i-1} \frac{|a_{ij}|}{|a_{ii}|} + \sum_{j=i+1}^{n} \frac{|a_{ij}|}{|a_{ii}|}\right) \max_{1\leqslant j\leqslant n} |z_j^k| = q \max_{1\leqslant j\leqslant n} |z_j^k|.$$

Отсюда следует, что

$$\|z^{k+1}\|_\infty \leqslant q\, \|z^k\|_\infty$$

для любого $k = 0, 1, \ldots$, поэтому

$$\|z^k\|_\infty \leqslant q\,\|z^{k-1}\|_\infty \leqslant q^2\,\|z^{k-2}\|_\infty \leqslant \ldots \leqslant q^k\|z^0\|_\infty \to 0$$

при $k \to \infty$, поскольку $0 < q < 1$, а это и означает, что $x^k \to x$. $\square$

Оценка (5) показывает, что, чем меньше q, т. е. чем выше диагональное преобладание матрицы A, тем быстрее сходится метод Якоби.

2. Метод Зейделя.[1)] Формулы (3) допускают естественную модификацию. Именно, при вычислении x_i^{k+1} будем использовать уже найденные компоненты вектора x^{k+1}, т. е. $x_1^{k+1}, x_2^{k+1}, \ldots x_{i-1}^{k+1}$. В результате приходим к итерационному *методу Зейделя*:

$$x_i^{k+1} = \Big(b_i - \sum_{j=1}^{i-1} a_{ij}\,x_j^{k+1} - \sum_{j=i+1}^{n} a_{ij}\,x_j^k\Big)/a_{ii}, \quad i = 1:n.$$

В матричных обозначениях он запишется в виде (проверьте!):

$$x^{k+1} = D^{-1}\big(b - Lx^{k+1} - Ux^k\big), \quad k = 0, 1, \ldots.$$

Метод Зейделя позволяет более экономно расходовать память компьютера, поскольку в данном случае вновь получаемые компоненты вектора x^{k+1} можно размещать на месте соответствующих компонент вектора x^k, в то время как при реализации метода Якоби все компоненты векторов x^k, x^{k+1} должны одновременно находиться в памяти ЭВМ.

Теорема 2. *Для матриц с диагональным преобладанием по строкам метод Зейделя линейно сходится при любом начальном приближении. Справедлива оценка скорости сходимости*

$$\|x - x^k\|_\infty \leqslant \rho^k\,\|x - x^0\|_\infty,$$

где $\rho \leqslant q$, q определяется формулой (4).

Доказательство. Аналогично методу Якоби имеем

$$z_i^{k+1} = -\sum_{j=1}^{i-1} \frac{a_{ij}}{a_{ii}}\,z_j^{k+1} - \sum_{j=i+1}^{n} \frac{a_{ij}}{a_{ii}}\,z_j^k, \quad i = 1:n. \tag{6}$$

[1)] Филипп Людвиг Зейдель (Philipp Ludwig von Seidel; 1821 — 1896) — немецкий математик и астроном.

Пусть $|z_l^{k+1}| = \max_{1\leqslant j\leqslant n} |z_j^{k+1}|$. Из l-того уравнения (6) следует, что

$$|z_l^{k+1}| \leqslant \alpha_l \max_{1\leqslant j\leqslant n} |z_j^{k+1}| + \beta_l \max_{1\leqslant j\leqslant n} |z_j^k|,$$

где

$$\alpha_l = \sum_{j=1}^{l-1} \frac{|a_{lj}|}{|a_{ll}|}, \quad \beta_l = \sum_{j=l+1}^{n} \frac{|a_{lj}|}{|a_{ll}|},$$

следовательно,

$$\|z^{k+1}\|_\infty \leqslant \frac{\beta_l}{1-\alpha_l} \|z^k\|_\infty.$$

Из условия (4) получаем, что $\alpha_l + \beta_l \leqslant q < 1$, т.е. и $q\alpha_l + \beta_l \leqslant q$. Таким образом, $\rho = \beta_l/(1-\alpha_l) \leqslant q$ и $\|z^{k+1}\|_\infty \leqslant \rho \|z^k\|_\infty$ для любого $k \geqslant 0$. Дальнейшие рассуждения совпадают с соответствующими рассуждениями из доказательства предыдущей теоремы. $\square$

3. Метод релаксации. Зачастую существенного ускорения сходимости можно добиться за счет введения в расчетные формулы числового параметра. В качестве примера приведем итерационный метод

$$x_i^{k+1} = (1-\omega)x_i^k + \omega\Big(b_i - \sum_{j=1}^{i-1} a_{ij}\, x_j^{k+1} - \sum_{j=i+1}^{n} a_{ij}\, x_j^k\Big)/a_{ii},$$

$i = 1:n$, $k = 0, 1, \ldots$ Этот метод называется *методом релаксации*, число ω — *релаксационным параметром*. При $\omega = 1$ метод переходит в метод Зейделя. В матричных обозначениях получаем

$$x^{k+1} = (1-\omega)x^k + \omega\, D^{-1}\big(b - Lx^{k+1} - Ux^k\big), \quad k = 0, 1, \ldots$$

Ясно, что по затратам памяти и объему вычислений на каждом шаге итераций метод релаксации не отличается от метода Зейделя.

§ 2. Элементы общей теории итерационных методов

1. Нормы. Далее наряду со стандартным скалярным произведением $(\cdot,\cdot)$ будем использовать *энергетическое скалярное произведение* и соответствующую ему норму на пространстве $\mathbb{R}^n$. Именно, если D — симметричная положительно определенная матрица, то по определению

$$(x,y)_D = (Dx, y), \quad \|x\|_D = (Dx, x)^{1/2}.$$

Через $\|T\|_D$ будем обозначать норму матрицы $T \in M^n$, подчиненную норме вектора $\|\cdot\|_D$.

2. Канонический вид простейших ИМ. Придадим итерационным методам, рассмотренным в предыдущих пунктах, удобные для анализа сходимости формулировки.

Начнем с метода Якоби. Нетрудно видеть, что равенства $x^{k+1} = D^{-1}\bigl(b - Lx^k - Ux^k\bigr)$ можно переписать в виде

$$D(x^{k+1} - x^k) + Ax^k = b,$$

где $D = \operatorname{diag}(a_{11}, a_{22}, \ldots, a_{nn})$. Простыми преобразованиями формулы $x^{k+1} = (1-\omega)x^k + \omega\, D^{-1}\bigl(b - Lx^{k+1} - Ux^k\bigr)$ метода релаксации могут быть переписаны как (проверьте! напомним, что $A = L + D + U$):

$$(D + \omega L)\,\frac{x^{k+1} - x^k}{\omega} + Ax^k = b.$$

Из этих формул, в частности, видно, что если ИМ сходятся, то они сходятся к решению исходной системы уравнений.

3. Канонический вид двухслойных ИМ. Будем рассматривать общий класс ИМ, определяемых соотношениями

$$B\,\frac{x^{k+1} - x^k}{\tau} + Ax^k = b, \quad k = 0, 1, \ldots, \tag{7}$$

где вектор x^0 считается заданным.[1)] Здесь B — невырожденная матрица, $\tau > 0$ — число, называемое *итерационным параметром*. При $B = I$ метод называется также *методом простой итерации*.

Для того, чтобы найти вектор x^{k+1} по уже известному вектору x^k, достаточно решить систему линейных уравнений

$$Bw^k = r^k, \tag{8}$$

где $r^k = Ax^k - b$, и положить $x^{k+1} = x^k - \tau w^k$. Отметим, что от A требуется уметь вычислять лишь произведение A на известный вектор x^k. Такая операция достаточно экономно реализуется даже для сверхбольших разреженных матриц.

[1)] Нетрудно доказать, что любой сходящийся линейный двухслойный ИМ может быть записан в виде (7). Поэтому он называется каноническим видом двухслойного ИМ.

Ясно, что при построении ИМ (7) матрица B должна выбираться так, чтобы решение системы уравнений вида (8) выполнялось намного быстрее, чем решение исходной системы $Ax = b$.

ИМ Якоби, Зейделя и релаксации являются частными случаями метода (7). Например, в случае метода Якоби $B = D$, $\tau = 1$.

4. Достаточные условия сходимости. Наша ближайшая цель — получить условия на матрицу B и параметр τ, обеспечивающие сходимость метода (7). Если x — решение системы $Ax = b$, то

$$B\,\frac{x - x}{\tau} + Ax = b. \tag{9}$$

Вычитая почленно равенства (9), (7), получим

$$B\,\frac{z^{k+1} - z^k}{\tau} + Az^k = 0,$$

откуда

$$z^{k+1} = Tz^k, \quad k = 0, 1, \ldots, \tag{10}$$

где

$$T = I - \tau B^{-1}A.$$

Таким образом, из (10) получаем $z^k = Tz^{k-1} = T^2z^{k-2}$ и т.д., т.е.

$$z^k = T^kz^0, \quad k = 0, 1, \ldots. \tag{11}$$

Понятно, что сходимость итерационного метода (7) полностью определяется свойствами матрицы T, которую обычно называют *матрицей перехода* (шага) итерационного метода (7).

Определение 2. *Пусть $\lambda_1(T)$, $\lambda_2(T)$, $\ldots$, $\lambda_n(T)$ — собственные числа матрицы T.* Спектральным радиусом T *называется число*

$$\rho(T) = \max_{1 \leqslant i \leqslant n} |\lambda_i(T)|.$$

Теорема 3. *Для того, чтобы итерационный метод* (7) *сходился при любом начальном приближении x^0, необходимо и достаточно, чтобы спектральный радиус $\rho(T)$ матрицы T был меньше единицы.*

Доказательство. Необходимость. Пусть λ — собственное число матрицы T такое, что $|\lambda| \geqslant 1$, e — соответствующий этому собственному

числу нормированный собственный вектор. Выберем в качестве начального приближения в итерационном методе (7) вектор $x^0 = x - e$, где x — решение системы $Ax = b$. Тогда $z^0 = e$ и в соответствии с (11) имеем $z^1 = Tz^0 = \lambda e$ и, вообще, $z^k = \lambda^k e$. Следовательно, $\|z^k\| = |\lambda|^k$. Очевидно, либо $\|z^k\| \to \infty$ при $k \to \infty$, либо $\|z^k\| = 1$ для всех $k \geqslant 1$, т.е. метод (7) не сходится.

Достаточность. Если спектральный радиус T меньше единицы, то она является сходящейся матрицей, т.е. $T^k \to 0$ при $k \to \infty$ (без доказательства). Тогда из (11) следует $z^k \to 0$ при $k \to \infty$. $\square$

Теорема 4. *Для сходимости ИМ* (7) *достаточно, чтобы выполнялось условие* $\|T\| < 1$ *для какой-либо подчиненной нормы матрицы. Справедлива оценка* $\|x - x^k\| \leqslant \|T\|^k \|x - x^0\|$, $k = 0, 1, \ldots$

Доказательство. Из равенства (11) следует искомая оценка

$$\|z^k\| = \|T^k z^0\| \leqslant \|T^k\| \, \|z^0\| \leqslant \|T\|^k \, \|z^0\|. \quad \square$$

Например, при $\tau = 1$ итерационный метод (7) сходится, если матрицы A и B достаточно близки, т. е. $\|B^{-1}\| \|B - A\| < 1$, поскольку в этом случае $T = B^{-1}(B - A)$.

Задания для самостоятельной работы

ВОПРОСЫ ДЛЯ САМОКОНТРОЛЯ

1. Какие методы решения СЛАУ называются: а) итерационными методами; б) двухслойными методами; с) трехслойными методами?

2. Из каких соображений выбирается начальное приближение в двухслойных методах?

3. Какой ИМ называется а) сходящимся; б) линейно сходящимся?

4. Приведите примеры условий завершения итераций.

5. В каких случаях ИМ могут быть предпочтительнее прямых методов?

6. Приведите формулы ИМ Якоби а) в матричном виде; б) индексном виде.

7. Приведите формулы ИМ Зейделя а) в матричном виде; б) индексном виде.

8. Приведите формулы ИМ релаксации а) в матричном виде; б) индексном виде.

9. Укажите достаточное условие сходимости метода Якоби и приведите оценку его скорости сходимости.

10. Укажите достаточное условие сходимости метода Зейделя и приведите оценку его скорости сходимости.

11. Укажите канонический вид двухслойных ИМ.

12. Запишите ИМ Якоби в каноническом виде. Как определяются матрица B и τ.

13. Запишите ИМ Зейделя в каноническом виде. Как определяются матрица B и τ.

14. Запишите ИМ релаксации в каноническом виде. Как определяются матрица B и τ.

15. Дайте определение а) энергетической нормы вектора; b) подчиненной нормы матрицы; c) спектрального радиуса матрицы.

16. Укажите вид матрицы перехода двухслойного ИМ. С какой целью эта матрица вводится?

17. Сформулируйте необходимое и достаточное условие сходимости двухслойного ИМ.

18. Сформулируйте достаточное условие сходимости двухслойного ИМ.

ЗАДАЧИ И УПРАЖНЕНИЯ

1. Пусть для решения системы $x = Tx + c$ используются итерации $x^{k+1} = Tx^k + c$,

$$T = \begin{pmatrix} \alpha & 1 \\ 0 & \beta \end{pmatrix}, \quad 0 < \beta < 1.$$

Найдите максимальное значение ρ такое, что ИМ сходится при произвольном начальном приближения при $|\alpha| < \rho$.

Указание. Получите матрицу перехода и воспользуйтесь необходимым и достаточным критерием сходимости.

2. Для решения системы $Ax = b$ с матрицей

$$A = \begin{pmatrix} \alpha & \beta & 0 \\ \beta & \alpha & \beta \\ 0 & \beta & \alpha \end{pmatrix}, \quad \alpha = \sqrt{2},$$

применяется итерационный метод Якоби. Найдите максимальное значение ρ такое, что он сходится при произвольном начальном приближения при $|\beta| < \rho$.

Указание. Запишите матрицу перехода и воспользуйтесь необходимым и достаточным критерием сходимости.

3. Для решения системы $Ax = b$ с матрицей

$$A = \begin{pmatrix} \alpha & \beta & 0 \\ \beta & \alpha & \beta \\ 0 & \beta & \alpha \end{pmatrix}, \quad \alpha = 2\sqrt{2},$$

применяется итерационный метод Зейделя. Найдите максимальное значение ρ такое, что он сходится при произвольном начальном приближения при $|\beta| < \rho$.

Указание. Запишите матрицу перехода и воспользуйтесь необходимым и достаточным критерием сходимости.

4. Найти максимальное значение ρ такое, что при $\tau \in (0, \rho)$ итерационный метод

$$\frac{x^{k+1} - x^k}{\tau} + Ax^k = b, \quad k = 0, 1, \ldots$$

для решения системы $Ax = b$ с матрицей

$$A = \begin{pmatrix} 1.5 & 0.5 \\ 0.5 & 1.5 \end{pmatrix}$$

сходится при произвольном начальном приближении x^0.

Указание. Запишите матрицу перехода и воспользуйтесь необходимым и достаточным критерием сходимости.

5. Для решения системы $x + Ax = b$ с матрицей

$$A = \begin{pmatrix} 0.01 & 0.06 & 0.03 \\ 0.02 & -0.05 & 0.02 \\ 0.01 & 0.03 & -0.04 \end{pmatrix}$$

применяется итерационный метод $x^{k+1} = Ax^k + b,\ k = 0, 1, 2, \ldots$. а) докажите, что ИМ сходится; б) найдите минимальное число итераций m, которое достаточно выполнить, чтобы гарантировать оценку точности $\max\limits_{1 \leqslant i \leqslant 3} |x_i - x_i^m| \leqslant 10^{-5} \max\limits_{1 \leqslant i \leqslant 3} |x_i - x_i^0|$.

Указание. Получите матрицу перехода T и оцените ее норму $\|T\|_\infty$.

§ 3. Достаточное условие сходимости при $A = A^T > 0$

Следующая теорема полезна при исследовании многих ИМ.[1)]

Теорема 5 (Самарский[2)]). *Пусть* $A = A^T > 0$ *и* $C = B - \tau/2\, A > 0$. *Тогда ИМ* (7) *сходится при любом начальном векторе. Кроме того, при любом* $k > 0$ *выполнено неравенство*

$$\|z^{k+1}\|_A < \|z^k\|_A, \quad z^k \neq 0. \tag{12}$$

Доказательство. Используя равенство $z^k = \frac{z^{k+1}+z^k}{2} - \frac{z^{k+1}-z^k}{2}$ уравнение для погрешности

$$B\,\frac{z^{k+1} - z^k}{\tau} + Az^k = 0, \tag{13}$$

преобразуем к виду

$$C\,\frac{z^{k+1} - z^k}{\tau} + \frac{1}{2}\,A(z^{k+1} + z^k) = 0. \tag{14}$$

Умножим обе части (14) скалярно на вектор $2(z^{k+1} - z^k)$ и преобразуем полученное с учетом симметрии матрицы A. Получим:

$$2/\tau\,\big(C(z^{k+1} - z^k), z^{k+1} - z^k\big) + (Az^{k+1}, z^{k+1}) - (Az^k, z^k) = 0. \tag{15}$$

По условию первое слагаемое в левой части (15) неотрицательно, поэтому $(Az^{k+1}, z^{k+1}) \leqslant (Az^k, z^k)$, т.е. числовая последовательность

[1)] Напомним, что $A > 0$ означает, что A есть положительно определенная матрица.

[2)] Александр Андреевич Самарский (1919 — 2008) — советский, российский математик.

$\|z^k\|_A$ невозрастает. Кроме того, она ограничена снизу нулем. Следовательно, последовательность (Az^k, z^k) имеет предел. Отсюда вытекает, что $(Az^{k+1}, z^{k+1}) - (Az^k, z^k) \to 0$ при $k \to \infty$, следовательно, и

$$\left(C\,(z^{k+1} - z^k), z^{k+1} - z^k\right) \to 0 \text{ при } k \to \infty,$$

а поскольку матрица $C > 0$, то $z^{k+1} - z^k \to 0$ при $k \to \infty$. Используя теперь уравнение (13) и невырожденность матрицы A, получим,

$$\|z^k\| = \|A^{-1}B\,\frac{z^{k+1} - z^k}{\tau}\| \leqslant \|A^{-1}B\|\,\|z^{k+1} - z^k\|/\tau \to 0$$

при $k \to \infty$. Теперь, если $z^k \neq 0$, то также $z^{k+1} - z^k \neq 0$ (почему?), и из (15) следует (12). $\square$

Используя эту теорему, исследуем сходимость метода релаксации.

Следствие 1. *Пусть матрица $A = A^T > 0$, параметр релаксации удовлетворяет условию $0 < \omega < 2$. Тогда метод релаксации сходится при любом начальном приближении x^0.*

Доказательство. Имеем $A = D+L+L^T$. Методу релаксации соответствуют $B = D+\omega L$, $\tau = \omega$, $C = B-\tau/2\,A = (1-\omega/2)D+\omega/2(L-L^T)$,

$$(Cx, x) = (1 - \omega/2)(Dx, x) + \omega/2((L - L^T)x, x).$$

Но все диагональные элементы положительно определенной матрицы положительны (докажите!), поэтому $(Dx, x) > 0$ при $x \neq 0$, а $(Lx, x) - (L^Tx, x) = (Lx, x) - (x, Lx) = 0$ для любого x. Т.о.,

$$(Cx, x) = (1 - \omega/2)(Dx, x) > 0 \quad \text{при} \quad x \neq 0,\ \omega \in (0, 2). \quad \square$$

Теорема 6. *Условие $\omega \in (0, 2)$ необходимо для сходимости метода релаксации при любом начальном приближении.*

Доказательство. Имеем $B = D + \omega L$. Запишем равенство $T = I - \tau B^{-1}A$ в виде $BT = B - \omega A$ или

$$(D + \omega L)T = (D + \omega L) - \omega A = (1 - \omega)D - \omega L^T. \qquad (16)$$

Поскольку L и L^T — строго треугольные матрицы, а D — диагональная матрица, все диагональные элементы которой отличны от нуля,

то, вычисляя определители левой и правой частей равенства (16), получим, что $\det(T) = (1-\omega)^n$, следовательно

$$\prod_{k=1}^{n} |\lambda_k| = |\det(T)| = |1-\omega|^n,$$

где $\lambda_1,\ \lambda_2,\ \ldots,\lambda_n$ — собственные числа матрицы T. Если условие $\omega \in (0,2)$ нарушено, то $|1-\omega| \geqslant 1$, и среди собственных чисел λ_k матрицы T есть хотя бы одно, модуль которого больше или равен единице, т.е. $\rho(T) \geqslant 1$. Но метод релаксации в этом случае не будет сходится при любом x^0. $\square$

§ 4. Оптимальный выбор итерационного параметра

Из доказательства необходимого и достаточного условия сходимости ИМ следовало, что ИМ (7) сходится тем быстрее, чем меньше спектральный радиус матрицы $T = I - \tau B^{-1}A$. В связи с этим возникает задача отыскания такого (*оптимального*) значения итерационного параметра τ, при котором величина $\rho(T)$ принимает минимальное значение.

Наиболее просто эта задача решается в случае, когда матрицы A, B симметричны и положительно определены.

Теорема 7. *Пусть A и B симметричные и положительно определенные матрицы; m минимальное, а M максимальное собственное число матрицы $B^{-1}A$. Тогда оптимальное значение итерационного параметра τ равно $\tau_0 = 2/(m+M)$, а для спектрального радиуса матрицы перехода T справедливо представление*

$$\rho(T) = \rho_0 = \frac{1-\xi}{1+\xi}, \quad \xi = \frac{m}{M}\,. \tag{17}$$

Доказательство. Пусть λ, x есть собственная пара матрицы $T = I - \tau B^{-1}A$, т.е. $Tx = \lambda x$. Тогда $BTx = \lambda Bx$ и

$$Bx - \tau Ax = \lambda Bx \quad \rightleftarrows \quad Ax = \mu\, Bx, \quad \mu = (1-\lambda)/\tau. \tag{18}$$

Поэтому

$$|\lambda| = |\,1 - \tau\, s\,|\,, \quad s = \frac{(Ax,x)}{(Bx,x)}. \tag{19}$$

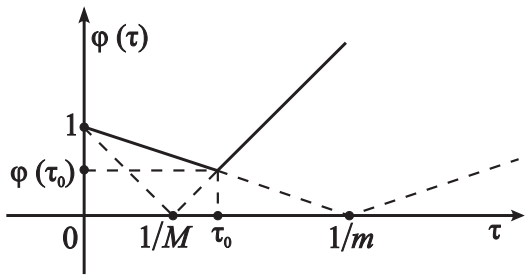

Рис. 1. К выбору оптимального итерационного параметра τ.

Ясно также, что x — собственный вектор матрицы $B^{-1}A$, точнее,

$$B^{-1}Ax = \mu\, x.$$

Очевидно, справедливо и обратное: любой собственный вектор $B^{-1}A$ есть собственный вектор матрицы T. Из (18) следует равенство $(Ax, x) = \mu(Bx, x)$. Поэтому все собственные числа $\mu > 0$. Тогда для любого собственного вектора x матрицы T справедливы неравенства

$$0 < m \leqslant \mu = \frac{(Ax, x)}{(Bx, x)} \leqslant M. \tag{20}$$

Полученные оценки являются точными, поскольку соответствующие неравенства (20) превращаются в равенства, если в качестве x взять собственный вектор, отвечающий m или M.

Функция $g(\mu) = |1 - \tau\mu|$ вещественного переменного μ на любом ограниченном отрезке вещественной оси достигает максимального значения на одном из концов этого отрезка. Поэтому, используя соотношения (19), (20), получаем, что

$$\rho(T) = \max_{1\leqslant i\leqslant n} |\lambda_i(T)| = \max_{m\leqslant s\leqslant M} |1 - \tau\, s| = \\ = \max\{|1 - \tau\, m|, |1 - \tau\, M|\} = \varphi(\tau).$$

График функции $\varphi(\tau)$ при $\tau \geqslant 0$ изображен на рис. 1. Имеем

$$\min_{\tau\geqslant 0} \varphi(\tau) = \varphi(\tau_0) = \rho_0 = \frac{M - m}{M + m} = \frac{1 - \xi}{1 + \xi},$$

где $\tau_0 = 2/(m + M)$, $\xi = m/M$. $\square$

Из теоремы следует, что ИМ (7) при оптимальном значении итерационного параметра $\tau = \tau_0$ сходится тем быстрее, чем больше ξ, т. е. чем меньше разброс собственных чисел матрицы $B^{-1}A$.

Получим оценку скорости сходимости метода (7). Для этого нам понадобится определение квадратного корня матрицы.

Дробная степень матрицы. Пусть $A = A^T \geqslant 0$. Из курса алгебры известно, что для любой симметричной матрицы справедливо спектральное разложение $A = U^T \Lambda U$, где U — ортогональная матрица, $\Lambda = \operatorname{diag}(\lambda_1(A), \ldots, \lambda_n(A))$ есть диагональная матрица с собственными числами A на диагонали. Т. к. $A \geqslant 0$, то все $\lambda_i(A) \geqslant 0$.

Лемма 1. *(самостоятельно!) Пусть* $\alpha \geqslant 0$, $A = U^T \Lambda U \geqslant 0$. *Определим матрицу* A^α *равенством*

$$A^\alpha = U^T \Lambda^\alpha U, \quad \Lambda^\alpha = \operatorname{diag}(\lambda_1^\alpha(A), \lambda_2^\alpha(A), \ldots, \lambda_n^\alpha(A)).$$

Тогда:

a) A^α *симметричная и неотрицательно определенная матрица, причем, если матрица* $A > 0$, *то и* $A^\alpha > 0$;

b) $A^\alpha A^\beta = A^{\alpha+\beta}$ *(в частности,* A^α *и* A *перестановочны, т. е.* $A^\alpha A = A A^\alpha$*).*

c) Положим $A^{-\alpha} = (A^\alpha)^{-1}$, *если* $A > 0$. *Тогда* $A^{-\alpha} = (A^{-1})^\alpha$.

Определение 3. *Матрицу* A^α *называют дробной степенью* A; *матрицу* $A^{1/2}$ *— корнем квадратным из матрицы* A.

Теорема 8. *Пусть матрицы* A, B *симметричны и положительно определены. Тогда для приближений, построенных по итерационному методу* (7) *при* $\tau = \tau_0 = 2/(m+M)$, *справедливы оценки*

$$\|x - x^k\|_A \leqslant \rho_0^k \, \|x - x^0\|_A, \quad k = 0, 1, \ldots,$$

где значение $\rho_0 = \rho(I - \tau_0 B^{-1} A)$ *определено в* (17), *т.е.*

$$\rho_0 = \frac{1-\xi}{1+\xi}, \quad \xi = \frac{m}{M},$$

а m, M *— min и max собственные числа матрицы* $B^{-1}A$.

Доказательство. После умножения равенства

$$z^{k+1} = T z^k = (I - \tau_0 \, B^{-1} A) z^k, \quad A = A^{1/2} A^{1/2},$$

на $A^{1/2}$ слева получим представление

$$A^{1/2} z^{k+1} = (I - \tau_0 \, A^{1/2} B^{-1} A^{1/2}) \, A^{1/2} z^k = S \, A^{1/2} z^k, \qquad (21)$$

где

$$S = I - \tau_0 A^{1/2} B^{-1} A^{1/2}.$$

Возьмем евклидову норму от обеих частей (21) и учтем, что $\|A^{1/2}z\|_2 = (A^{1/2}z, A^{1/2}z)^{1/2} = (Az, z)^{1/2} = \|z\|_A$. Получим

$$\|z^{k+1}\|_A = \|S\, A^{1/2} z^k\|_2 \leqslant \|S\|_2 \|A^{1/2} z^k\|_2 = \|S\|_2 \|z^k\|_A.$$

Матрица S, очевидно, симметрична, поэтому $\|S\|_2 = \rho(S)$. Пусть y, λ — собственная пара матрицы $A^{1/2}B^{-1}A^{1/2}$, т. е.

$$A^{1/2} B^{-1} A^{1/2} y = \lambda y. \tag{22}$$

Полагая $y = A^{-1/2}Bz$, получим, что собственные значения задачи (22) совпадают с собственными значениями матрицы $B^{-1}A$. Отсюда следует, что $\rho(S) = \rho_0$ (см. доказательство теоремы 7). Следовательно, $\|z^{k+1}\|_A \leqslant \rho_0 \|z^k\|_A$. $\square$

§ 5. Критерии остановки итераций и выбор матрицы B

1. Критерии остановки итераций. ИМ генерирует бесконечную последовательность приближений $\{x^k\}_{k=0}^n$, сходящуюся к решению x. Зная дополнительно матрицу A и правую часть b, мы должны решить, когда прервать этот процесс. Идеальное решение: когда будет выполнено условие $\|x - x^k\| \leqslant \varepsilon$ или, что лучше, $\|x - x^k\|/\|x\| \leqslant \varepsilon$ для заданного ε. Однако этот критерий окончания ИМ практически не пригоден, поскольку решение x нам не известно.

На практике о сходимости судят по малости величин $\|x^k - x^{k-1}\|$ или $\|r^k\|$, где $r^k = Ax^k - b$ — невязка приближения x^k. Интуитивно ясно, что малость $\|x^k - x^{k-1}\|$ влечет близость к решению только для достаточно быстро сходящихся последовательностей. Это подтверждает следующая

Лемма 2. *Пусть известно $q < 1$ такое, что $\|x - x^k\| \leqslant q\,\|x - x^{k-1}\|$ для $k = 1, 2, \ldots$. Тогда*

$$\|x - x^k\| \leqslant Q\,\|x^k - x^{k-1}\|, \quad Q = \frac{q}{1-q} > q.$$

Доказательство. Имеем, очевидно,

$$\|x - x^{k-1}\| \leqslant \|x - x^k\| + \|x^k - x^{k-1}\| \leqslant q\,\|x - x^{k-1}\| + \|x^k - x^{k-1}\|.$$

Отсюда следует, что $\|x-x^{k-1}\| \leqslant 1/(1-q)\,\|x^k-x^{k-1}\|$. Следовательно,

$$\|x - x^k\| \leqslant q\,\|x - x^{k-1}\| \leqslant Q\,\|x^k - x^{k-1}\|. \quad \square$$

Из этой леммы следует, что

$$\|x^k - x^{k-1}\| \leqslant \varepsilon/Q \quad \Rightarrow \quad \|x - x^k\| \leqslant \varepsilon.$$

Для медленно сходящихся последовательностей имеем $q \approx 1$ и $Q \gg 1$. Для быстро сходящихся последовательностей Q не велико. Например, при $q \approx 0.5$, имеем $Q \approx 1$.

Лемма 3. *Пусть известно Q такое, что $\|A^{-1}\| \leqslant Q$ для некоторой подчиненной нормы матрицы. Тогда $\|x - x^k\| \leqslant Q\,\|r^k\|$ для $k \geqslant 0$.*

Доказательство. Имеем, очевидно, $A(x^k - x) = Ax^k - b = r^k$. Следовательно, $x^k - x = A^{-1}r^k$ и $\|x^k - x\| \leqslant \|A^{-1}\|\,\|r^k\|$. $\square$

Из этой леммы следует, что

$$\|r^k\| \leqslant \varepsilon/Q \quad \Rightarrow \quad \|x - x^k\| \leqslant \varepsilon.$$

Отметим случай $A = A^T > 0$ и $\|x\| = \|x\|_2$. Тогда $\|A^{-1}\| = 1/\lambda_{\min}(A)$. Поэтому, если $\lambda_{\min}(A) \geqslant \delta$, то $\|A^{-1}\| \leqslant 1/\delta = Q$ и Q тем больше, чем ближе с.ч. A расположены к нулю.

2. Выбор матрицы B. Матрицу B часто называют матрицей предобуславливания СЛАУ или просто предобуславливателем. Ее выбор очень важен, особенно при решении больших разреженных СЛАУ. Отметим лишь два способа его выбора.

1. Предобуславливатель Якоби. В этом случае B — диагональная матрица, $B = D = \operatorname{diag}(d_1, d_2, \ldots, d_n)$. Выбор в качестве D диагонали матрицы A является подходящим, если $A = A^T > 0$. В случае нессимметричной A можно принять $d_i = \left(\sum_{j=1}^n a_{ij}^2\right)^{1/2}$.

2. Предобуславливатель SSOR в случае $A = A^T > 0$. Пусть $A = L + D + L^T$ есть разложение A на строго нижнюю треугольную, диагональную и верхнюю треугольные матрицы, $\omega \in (0, 2)$. SSOR предобуславливатель определяется формулой

$$B = \left(\frac{1}{\omega}D + L\right)\left(\frac{1}{\omega}D\right)^{-1}\left(\frac{1}{\omega}D + L\right)^T.$$

Задания для самостоятельной работы

ВОПРОСЫ ДЛЯ САМОКОНТРОЛЯ

1. Укажите достаточный критерий сходимости двухслойного ИМ в каноническом виде при $A = A^T > 0$.

2. Приведите формулу для оптимального итерационного параметра в случае симметричных и положительно-определенных матриц A и B.

3. Приведите оценку скорости сходимости ИМ при оптимальном параметре в случае симметричных и положительно-определенных матриц A и B.

4. Дайте определение дробной степени матрицы. Для каких матриц она определяется?

5. Укажите способы выбора матрицы предобуславливания B.

ЗАДАЧИ И УПРАЖНЕНИЯ

1. При каких α матрица

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

а) имеет диагональное преобладание?
б) является положительно определенной (т.е. $A > 0$)?
с) имеет положительные собственные числа?

2. Найдите нормы $\|A\|_1$, $\|A\|_2$, $\|A\|_\infty$ матрицы

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

3. Найти максимальное значение ρ такое, что при $\tau \in (0, \rho)$ итерационный метод

$$\frac{x^{k+1} - x^k}{\tau} + Ax^k = b, \quad k = 0, 1, \ldots$$

для решения системы $Ax = b$ с матрицей

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

сходится при произвольном начальном приближении x^0. Используйте следующие критерии сходимости ИМ:
а) необходимый и достаточный критерий.
b) критерий $B - \tau/2A > 0$.
с) критерий $\|T\|_\infty < 1$.
d) критерий $\|T\|_2 < 1$.

Какой критерий дает большее значение ρ? О чем это говорит?

4. Найдите квадратный корень $A^{1/2}$ матрицы

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

Проверьте, что на самом деле $A = A^{1/2}A^{1/2}$.

5. Докажите, что если выполнены условия теоремы 8, то справедливы оценки

$$\|x - x^k\|_B \leqslant \rho_0^k \|x - x^0\|_B, \quad k = 1, 2, \ldots$$

6 Докажите, что если матрицы A, B симметричны и положительно определены, то итерационный метод (7) сходится при любом $\tau \in (0, 2/M)$.

7. Докажите следующую теорему.

Теорема 9. *Пусть A и B симметричные и положительно определеные матрицы, положительные числа m и M таковы, что $m(Bx, x) \leqslant (Ax, x) \leqslant M(Bx, x)$ для любого $x \in \mathbb{R}^n$. Тогда оптимальное значение итерационного параметра τ равно $\tau_0 = 2/(m + M)$, а для спектрального радиуса матрицы перехода $T = I - \tau_0 B^{-1} A$ справедлива оценка*

$$\rho(T) \leqslant \frac{1 - \xi}{1 + \xi}, \quad \xi = \frac{m}{M}.$$

Указание: Повторите доказательство аналогичной теоремы 3, внося малые поправки. Обратите внимание, что оценка (20) справедлива для всех x, а не только для собственных векторов. Обратите также внимание, что теперь m (M) есть оценка снизу (сверху) для минимального (максимального) собственного числа матрицы $B^{-1}A$.

8. Докажите следующую теорему.

Теорема 10. *Пусть A и B симметричные и положительно определеные матрицы, положительные числа m и M таковы, что $m(Bx, x) \leqslant (Ax, x) \leqslant M(Bx, x)$ для любого $x \in \mathbb{R}^n$. Тогда для приближений, построенных по итерационному методу* (7) *при $\tau = \tau_0 = 2/(m + M)$, справедливы следующие оценки:*

$$\|x - x^k\|_A \leqslant \rho_0^k \|x - x^0\|_A, \quad k = 0, 1, \ldots,$$

где

$$\rho_0 = \frac{1 - \xi}{1 + \xi}, \quad \xi = \frac{m}{M}.$$

Указание: Повторите доказательство аналогичной теоремы 8, внося малые поправки.

§ 6. Итерационные методы вариационного типа

Отыскание оптимального значения параметра τ в рассмотренных ранее методах требует предварительного вычисления минимального и максимального собственных чисел матрицы $B^{-1}A$ или их оценок. Существуют итерационные методы, позволяющие за счет некоторой дополнительной работы на каждом шаге итераций автоматически настраиваться на оптимальную скорость сходимости. К их числу относятся методы, основанные на замене системы $Ax = b$ эквивалентной задачей минимизации некоторой квадратичной функции n переменных. Докажите самостоятельно следующее утверждение.

Лемма 4. *Пусть A невырожденная матрица. Тогда решение системы $Ax = b$ эквивалентно задаче: найти $x \in \mathbb{R}^n$ такой, что*

$$F(x) = \min_{y \in \mathbb{R}^n} F(y), \quad F(y) = \|Ay - b\|^2.$$

В важном случае можно взять "лучшую" функцию (см. упр. 1).

Теорема 11. *Пусть матрица $A = A^T > 0$. Тогда задача $Ax = b$ эквивалентна задаче отыскания минимума квадратичной функции*

$$F(y) = (Ay, y) - 2(b, y), \quad y \in \mathbb{R}^n.$$

Доказательство. Пусть x — решение системы $Ax = b$. Используя симметрию матрицы A, получим

$$F(y) = (Ay, y) - 2(Ax, y) + (Ax, x) - (Ax, x) = \\ = (A(y - x), y - x) - (Ax, x) = \|y - x\|_A^2 - \|x\|_A^2, \quad (23)$$

то есть минимума функция $F(y)$ достигает только при $y = x$. $\square$

Различные методы минимизации функции $F(y)$ приводят к различным ИМ для системы уравнений $Ax = b$. Далее мы ограничимся лишь рассмотрением случая $A = A^T > 0$ и функции

$$F(y) = (Ay, y) - 2(b, y) = \sum_{i,j=1}^{n} a_{ij}\, y_j\, y_i - 2 \sum_{i=1}^{n} b_i\, y_i.$$

1. Метод покоординатного спуска. Выберем некоторое начальное приближение $x^0 = (x_1^0, \ldots, x_n^0)^T$ и найдем величину x_1^1, доставляющую минимальное значение функции одной переменной $F(y_1, x_2^0, \ldots, x_n^0)$. Затем рассмотрим функцию одной переменной $F(x_1^1, y_2, x_3^0, \ldots, x_n^0)$ и найдем точку x_2^1, в которой она достигает минимума. Выполнив n таких шагов, построим вектор $x^1 = (x_1^1, \ldots, x_n^1)^T$ и примем его за новое приближение. Описанный процесс повторим, отталкиваясь от x^1 вместо x^0, получим x^2 и т.д.

Используя конкретный вид функции $F(x)$, найдем явные формулы для вычисления векторов x^k, $k = 1, 2 \ldots$ в полученном итерационном процессе. Компонента x_i^{k+1} вектора x^{k+1} разыскивается как точка минимума функции $F(x_1^{k+1}, \ldots, x_{i-1}^{k+1}, y_i, x_{i+1}^k, \ldots, x_n^k)$. Выпишем необходимое условие экстремума:

$$F'_{y_i}(x_1^{k+1}, \ldots, x_{i-1}^{k+1}, y_i, x_{i+1}^k, \ldots, x_n^k) = 0. \quad (24)$$

Вычисляя производную функции $F(y)$ по переменной y_i, получим:

$$F'_{y_i}(y) = 2\sum_{j=1}^{n} a_{ij}y_j - 2b_i, \tag{25}$$

следовательно, решая уравнение (24) относительно y_i, будем иметь

$$x_i^{k+1} = \Big(b_i - \sum_{j=1}^{i-1} a_{ij}\, x_j^{k+1} - \sum_{j=i+1}^{n} a_{ij}\, x_j^k\Big)/a_{ii}\,, \quad i = 1:n,$$

т.е. метод покоординатного спуска совпал с методом Зейделя.

2. Метод релаксации. Метод релаксации также допускает аналогичную геометрическую интерпретацию. При $\omega < 1$ из точки

$$(x_1^{k+1}, \ldots, x_{i-1}^{k+1}, x_i^k, x_{i+1}^k, \ldots, x_n^k)$$

двигаются в направлении координатной оси y_i, не доходя до точки минимума функции F на этой прямой, а при $\omega > 1$ проходят несколько дальше, чем точка минимума функции. Во многих случаях последний способ приводит к ускорению сходимости.

3. Метод наискорейшего спуска. Будем минимизировать функцию $F(y) = (Ay, y) - 2(b, y)$ на $\mathbb{R}^n$. Напомним, что также

$$F(y) = \|y - x\|_A^2 - \|x\|_A^2, \tag{26}$$

где x есть решение системы $Ax = b$.

Пусть приближение x^k задано. Будем двигаться из точки x^k в направлении наибыстрейшего убывания функции F пока не достигнем ее минимального значения, т. е. следующее приближение разыскиваем в виде $x^{k+1} = x^k - \tau$ grad $F(x^k)$, где

$$\text{grad } F(y) = (F'_{y_1}(y), F'_{y_2}(y), \ldots, F'_{y_n}(y))^T.$$

Из формулы (25) следует, что grad $F(x^k) = 2(Ax^k - b)$. Вектор $r^k = Ax^k - b$ принято называть вектором невязки. Для сокращения записей удобно обозначить 2τ вновь через τ. Таким образом, $x^{k+1} = x^k - \tau\, r^k$.

Параметр τ выберем так, чтобы значение $F(x^{k+1})$ было минимальным. Проводя элементарные выкладки, получим

$$F(x^{k+1}) = F(x^k - \tau r^k) = F(x^k) - 2\tau(r^k, r^k) + \tau^2(Ar^k, r^k),$$

следовательно, минимум $F(x^{k+1})$ достигается при $\tau = \tau_k = (r^k, r^k)/(Ar^k, r^k)$. Таким образом, мы пришли к следующему итерационному методу

$$x^{k+1} = x^k - \tau_k r^k, \quad r^k = Ax^k - b, \ \tau_k = \frac{(r^k, r^k)}{(Ar^k, r^k)}, \quad k = 0, 1, \ldots \quad (27)$$

Этот метод называется *методом наискорейшего спуска* (МНС). Первое равенство в (27) можно записать в виде

$$\frac{x^{k+1} - x^k}{\tau_k} + Ax^k = b, \quad k = 0, 1, \ldots$$

что совпадает по форме с каноническим видом двухслойного ИМ при $B = I$, но при переменном итерационном параметре. Его вычисление требует дополнительных арифметических операций. Вследствие этого происходит адаптация метода к оптимальной скорости сходимости.

При программировании МНС по указанным выше формулам требуется два умножения матрицы A на вектор, т.к. требуется вычислять векторы Ax^k и Ar^k. Одно вычисление матрицы на вектор можно исключить, если использовать следующее соображение. После умножения равенства $x^{k+1} = x^k - \tau_k r^k$ на матрицу A получаем, что векторы невязки также можно перевычислять на итерациях:

$$Ax^{k+1} = Ax^k - \tau_k Ar^k \quad \leftrightarrows \quad r^{k+1} = r^k - \tau_k Ar^k.$$

Это приводит к следующему методу, который и программируется:

1) задается x^0 и вычисляется $r^0 = Ax^0 - b$.
2) для $k = 0, 1, \ldots$ выполняются операции
   - $\diamond\ p^k = Ar^k$
   - $\diamond\ \tau_k = (r^k, r^k)/(p^k, r^k)$
   - $\diamond\ r^{k+1} = r^k - \tau_k p^k$
   - $\diamond\ x^{k+1} = x^k - \tau_k r^k$

   до выполнения условия $\|r^{k+1}\| \leqslant \varepsilon \|b\|$.

Теорема 12. *Пусть $A = A^T > 0$, m (M) ее минимальное (максимальное) собственное число. Тогда метод наискорейшего спуска сходится при любом начальном приближении и справедлива оценка*

$$\|x - x^k\|_A \leqslant \rho_0^k \|x - x^0\|_A, \quad k = 0, 1, \ldots, \quad (28)$$

где

$$\rho_0 = \frac{1-\xi}{1+\xi}, \quad \xi = \frac{m}{M} = \frac{1}{\mathrm{cond}_2(A)}\,.$$

Доказательство. Имеем по определению для любого τ_0

$$F(x^{k+1}) = F(x^k - \tau_k r^k) = \min_{\tau>0} F(x^k - \tau r^k) \leqslant F(x^k - \tau_0 r^k). \qquad (29)$$

Выберем $\tau_0 = 2/(m+M)$. Используем представление (26) в левой и правой части (29). Придем к оценке

$$\|x^{k+1} - x\|_A \leqslant \|x^k - \tau_0\, r^k - x\|_A. \qquad (30)$$

Заметим теперь, что ($z^k = x^k - x$, $b = Ax$)

$$x^k - \tau_0\, r^k - x = z^k - \tau_0\,(Ax^k - b) = z^k - \tau_0\, Az^k = Tz^k,$$

где $T = I - \tau_0 A$. Поэтому из (30) получим

$$\|z^{k+1}\|_A \leqslant \|Tz^k\|_A \leqslant \|T\|_A \|z^k\|_A.$$

В силу симметрии T и перестановочности T и A, нетрудно доказывается, что $\|T\|_A = \|T\|_2 = \rho(T)$. При оптимальном параметре τ_0 (в случае $B = I$) нами было доказано, что $\rho(T) = \rho_0$. Т.о. $\|z^{k+1}\|_A \leqslant \rho_0\, \|z^k\|_A$. $\square$

Замечание 1. Из (28) следует, что погрешность метода убывает с той же скоростью, что и погрешность метода простой итерации при оптимальном τ.

4. Предобусловленный вариант МНС. Из теоремы (12) следует, что скорость сходимости МНС тем хуже, чем хуже обусловлена матрица A. Для плохо обусловленных матриц $\mathrm{cond}_2(A) \gg 1$, т.е. $\xi \ll 1$ и $\rho_0 \approx 1$. Следовательно, число итераций, необходимых для достижения требуемой точности может быть недопустимо велико. С целью сокращения числа итераций в МНС вводят матричный параметр B исходя их следующих ограничений:

1) $B = B^T > 0$;
2) система уравнений $Bw = r$ просто (экономично) решается;
3) $\frac{\lambda_{min}(B^{-1}A)}{\lambda_{max}(B^{-1}A)} \gg \frac{\lambda_{min}(A)}{\lambda_{max}(A)}$.

Пусть B — симметричная положительно определенная матрица. Преобразуем систему $Ax = b$ к следующему виду:

$$B^{-1/2} A B^{-1/2} B^{1/2} x = B^{-1/2} b.$$

Полагая

$$C = B^{-1/2} A B^{-1/2}, \quad y = B^{1/2} x, \quad f = B^{-1/2} b, \tag{31}$$

получим новую систему

$$Cy = f. \tag{32}$$

Матрица C симметрична и положительно определена (докажите!).

Запишем формулы метода наискорейшего спуска и соответствующую оценку погрешности применительно к уравнению (32):

$$y^{k+1} = y^k - \tau_k \, (Cy^k - f), \quad k = 0, 1, \ldots, \tag{33}$$

$$\tau_k = \frac{(R^k, R^k)}{(CR^k, R^k)}, \tag{34}$$

где

$$R^k = Cy^k - f, \tag{35}$$

$$\|y^k - y\|_C \leqslant \rho_0^k(C) \, \|y^0 - y\|_C, \quad k = 1, 2, \ldots, \tag{36}$$

$$\rho_0(C) = \frac{1 - \xi}{1 + \xi}, \quad \xi = \frac{m_C}{M_C} = \frac{1}{\operatorname{cond}_2(C)} \, .$$

M_C, m_C — max и min собственные числа матрицы C соответственно.

На первый взгляд, полученные формулы кажутся практически бесполезными, так как матрица C и вектор f не определены конструктивно. Тем не менее, формулы удачно преобразуются.

Умножим обе части уравнения (33) на $B^{-1/2}$ и воспользуемся затем равенствами (31). В результате получим ИМ

$$x^{k+1} = x^k - \tau_k \, w^k, \quad k = 0, 1, \ldots, \tag{37}$$

где $w^k = B^{-1} r^k$, $r^k = Ax^k - b$, $x^k = B^{-1/2} y^k$. Далее, преобразуем (34) принимая во внимание (31), (35). Получим

$$\tau_k = \frac{(w^k, r^k)}{(Aw^k, w^k)}, \quad k = 0, 1, \ldots$$

Наконец, аналогичные преобразования приводят оценку (36) к виду

$$\|x^k - x\|_A \leqslant \rho_0^k(C) \, \|x^0 - x\|_A, \quad k = 1, 2, \ldots \tag{38}$$

Как показывает оценка (38), скорость сходимости метода (37) определяется собственными числами задачи $Cy = \lambda y$. Более подробная ее запись с использованием (31) дает $B^{-1/2}AB^{-1/2}y = \lambda y$. Матрица $B^{-1/2}$ обратима, поэтому, полагая $B^{-1/2}y = x$, приходим к эквивалентной задаче $Ax = \lambda Bx$, уже рассматривавшейся нами ранее. Поэтому величина ξ в определении $\rho_0(C)$ в оценке (38) равна

$$\xi = \frac{\lambda_{min}(B^{-1}A)}{\lambda_{max}(B^{-1}A)}.$$

Основная задача выбора B, т.о. формулируется так: выбрать достаточно простую матрицу $B = B^T > 0$ так, чтобы увеличить ξ.

Метод, описанный в настоящем пункте, часто называют *предобусловленным* методом наискорейшего спуска (ПМНС). Перевычисляя невязки на итерациях, аналогично МНС, придем к методу:

1) задается x^0 и вычисляется $r^0 = Ax^0 - b$.
2) для $k = 0, 1, \ldots$ выполняются операции
   - $\diamond$ решается система $Bw^k = r^k$
   - $\diamond$ $p^k = Aw^k$
   - $\diamond$ $\tau_k = (w^k, r^k)/(p^k, w^k)$
   - $\diamond$ $r^{k+1} = r^k - \tau_k\, p^k$
   - $\diamond$ $x^{k+1} = x^k - \tau_k w^k$

   до выполнения условия $\|r^{k+1}\| \leqslant \varepsilon\, \|b\|$.

При программировании индексы k и $k+1$ здесь можно опустить.

§ 7. Метод сопряженных градиентов.

Метод сопряженных градиентов (CG метод) является наиболее популярным и эффективным ИМ решения СЛАУ $Ax = b$ с симметричной и положительно определенной матрицей A. Он также может рассматриваться как метод минимизации квадратичной функции $F(y) = (Ay, y) - 2(b, y)$.

Этот метод относится к трехслойным методам и в точной арифметике сходится к решению x не более чем за n итераций. Это следует из того, что в этом методе приближения x^k строятся так, что невязки $r^k = b - Ax^k$ (я сменил знак!) оказываются ортогональными, а направления спуска p^k — A-ортогональными, т.е. для любых

$k, \ell = 0, 1, \ldots, n-1$, при $k \neq \ell$ выполняются соотношения

$$(r^k, r^\ell) = 0, \quad (Ap^k, p^\ell) = 0.$$

В $\mathbb{R}^n$ не может быть больше чем n таких ненулевых векторов и, поэтому, $r^m = 0$ при некотором $m \leqslant n$ и x^m есть точное решение СЛАУ.

Однако на практике, при наличии ошибок округления (при вычислениях на ЭВМ), эти вектора теряют ортогональность и метод становится итерационным. Имеется множество формул, описывающих CG-метод. Они эквивалентны в точной арифметике, но приводят к разным алгоритмам при наличии ошибок округления. Опишем один алгоритм, записанный в так называемой форме сопряженных направлений.

Первый шаг CG метода совпадает с первым шагом МНС. Далее приближения к решению x^k генерируются в направлении спуска p^k:

$$x^k = x^{k-1} + \alpha_k p^k.$$

Соответственно, невязки пересчитываются по правилу

$$r^k = r^{k-1} - \alpha_k\, q^k, \quad \text{где} \;\; q^k = Ap^k.$$

Выбор $\alpha_k = (r^{k-1}, r^{k-1})/(Ap^k, p^k)$ минимизирует F в направлении спуска $x^{k-1} + \alpha\, p^k$, т.е. $F(x^k) = \min_{\alpha>0} F(x^{k-1} + \alpha\, p^k)$.

Направление спуска p^k пересчитывается, используя невязки:

$$p^k = r^k + \beta_{k-1} p^{k-1}, \quad \beta_{k-1} = \frac{(r^{k-1}, r^{k-1})}{(r^{k-2}, r^{k-2})},$$

а выбор β_{k-1} обеспечивает ортогональность p^k и Ap^{k-1} или, эквивалентно, ортогональность r^k и r^{k-1}. Оказывается, что такой выбор β_{k-1} также обеспечивает ортогональность p^k и r^k всем ранее построенным Ap^j и r^j, соответственно.

Как и МНС CG-метод допускает предобуславливание заданием матрицы $B = B^T > 0$. Справедлива оценка скорости сходимости:

$$\|x - x^k\|_A \leqslant 2\rho_0^k \, \|x - x^0\|_A, \quad k = 1, 2, \ldots,$$

где

$$\rho_0 = \frac{1 - \sqrt{\xi}}{1 + \sqrt{\xi}}, \quad \xi = \frac{\lambda_{\min}(B^{-1}A)}{\lambda_{\max}(B^{-1}A)}.$$

Отличие от оценки ПМНС лишь в том, что в определении ρ_0 величина ξ заменена на $\sqrt{\xi}$. Это существенно лучше, т.к. $\xi < 1$.

Метод сопряженных градиентов (PCG).

1) задается $x = x^0$ и вычисляется $r = b - Ax$, $p = B^{-1}r$, $res = (r, p)$, $res0 = \varepsilon\,(b, b)$.

2) для $k = 1, 2, \ldots$ выполняются операции

- $\diamond\ q = Ap$
- $\diamond\ \alpha = res/(p, q)$
- $\diamond\ x = x + \alpha\, p$
- $\diamond\ r = r - \alpha\, q$
- $\diamond$ решается система $Bs = r$
- $\diamond\ resold = res$
- $\diamond\ res = (r, s)$
- $\diamond\ \beta = res/resold$
- $\diamond\ p = s + \beta\, p$

до выполнения условия $res < res0$.

Как видим, трудоемкость метода на итерации складывается из: 1) одного умножения матрицы A на заданный вектор; 2) решения системы $Bw = r$; 3) двух вычислений скалярного произведения; 4) трех операций вида $ax + y$, где $a \in \mathbb{R}$, $x, y \in \mathbb{R}^n$. Кроме памяти требуемой для реализации шагов 1) и 2), требуется дополнительно хранить в памяти ЭВМ 6 векторов длины n.

Задания для самостоятельной работы

ВОПРОСЫ ДЛЯ САМОКОНТРОЛЯ

1. Какой задаче минимизации эквивалентна задача решения системы $Ax = b$ в случае произвольной невырожденной матрицы A?

2. Какой задаче минимизации эквивалентна задача решения системы $Ax = b$ в случае $A = A^T > 0$?

3. В состоит идея метода покоординатного спуска?

4. Дайте определение метода наискорейшего спуска. а) Как выбирается направление спуска? б) Из каких соображений определяется шаг спуска?

5. Приведите оценку скорости сходимости метода наискорейшего спуска. а) Для каких матриц A метод быстро сходится? б) Для каких матриц A метод медленно сходится?

6. С какой целью в метод наискорейшего спуска вводится матричный параметр B и определяется ПМНС?

7. Приведите оценку скорости сходимости ПМНС. Из каких соображений необходимо выбирать предобуславливатель B?

8. Приведите оценку скорости сходимости метода PCG. Какой метод быстрее сходится: ПМНС или PCG и почему?

ЗАДАЧИ И УПРАЖНЕНИЯ

1. Вычислите градиент функции $F(x) = x_1^2 + 2x_1x_2 + x_2^2 + 3x_1 + x_2 - 1$ в точке $x = (1, 1)^T$.

2. Цель этого упражнения в том, чтобы продемонстрировать, что в случае $A = A^T > 0$ задаче минимизации

$$\Phi(x) = \min_{y \in \mathbb{R}^n} \Phi(y), \quad \Phi(y) = \|Ay - b\|^2,$$

лучше предпочесть минимизацию функции

$$F(y) = (Ay, y) - 2(b, y), \quad y \in \mathbb{R}^n,$$

хотя точка минимума обеих функций совпадает с решением системы $Ax = b$.

а) Докажите, что необходимое условие минимума Φ приводит к СЛАУ $A^TAx = A^Tb$ (симметризация Гаусса системы $Ax = b$).

б) Докажите, что $\Phi(y) = (A^TAy, y) - 2(A^Tb, y) + \|b\|^2 = (\bar{A}y, y) - 2(\bar{b}, y) + \|b\|^2$, где $\bar{A} = A^TA$, $\bar{b} = A^Tb$.

с) Докажите, что $\mathrm{cond}_2(\bar{A}) = (\mathrm{cond}_2(A))^2$.

Из б) следует, что минимизация Φ равносильна минимизации F, но с матрицей $\bar{A}$ вместо A, которая, в силу с), существенно хуже обусловлена.

3. Определите трудоемкость одной итерации: а) ПМНС; b) PCG. Принять, что трудоемкость решения системы с матрицей B равна двукратной трудоемкости умножения матрицы A на заданный вектор и равна mn, $m \geqslant 1$.

4. Пусть для решения системы $Ax = b$ с матрицей

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

используется а) МНС. Получите оценку скорости сходимости этого метода; б) метод PCG с матрицей $B = I$. Получите оценку скорости сходимости этого метода. с) какой из этих методов сходится быстрее?

ГЛАВА 3

Алгебраическая проблема собственных значений

Под алгебраической проблемой собственных значений понимают задачу отыскания собственных значений (с.з.) и собственных векторов (с.в.) квадратной матрицы. Различают *частичную* и *полную проблему* с.з. В первом случае необходимо определить лишь некоторые из с.з. и, возможно, соответствующие им с.в. Во втором случае ставится задача отыскания всех с.з. и, возможно, всех с.в.

Понятно, что методы решения частичной проблемы должны быть более простыми и менее трудоемкими. Мы рассмотрим примеры методов обоих классов. При этом ограничимся лишь случаем симметричных матриц. В этом случае все с.з. и с.в. являются вещественными.

1. Замечание о численных методах. Пусть задана вещественная квадратная матрица A размера n.

По определению число λ и ненулевой вектор x такие, что

$$Ax = \lambda x \quad \rightleftharpoons \quad (A - \lambda\, I)\, x = 0, \tag{1}$$

называются с.з. и с.в. матрицы A (соответствующим λ). Пара (λ, x) называется также собственной парой A.

Поскольку $x \neq 0$, то из второго равенства (1) следует, что λ является корнем характеристического полинома

$$P_n(\lambda) = \det\,(A - \lambda I) = (-1)^n(\lambda^n + p_1\,\lambda^{n-1} + p_2\,\lambda^{n-2} + \ldots + p_n).$$

Поскольку такой полином имеет ровно n корней (с учетом кратности), то матрица A также имеет ровно n с.з. (с учетом кратности).

В докомпьютерную эпоху методам вычисления коэффициентов характеристического полинома уделялось большое внимание. Изящный способ получения характеристического многочлена дал А. М. Данилевский. Его метод основан на преобразовании уравнения $P_n(\lambda) =$

$\det(A - \lambda I) = 0$ к виду

$$\begin{vmatrix} -p_1 - \lambda & -p_2 & -p_3 & \dots & -p_{n-1} & -p_n \\ 1 & -\lambda & 0 & \dots & 0 & 0 \\ 0 & 1 & -\lambda & \dots & 0 & 0 \\ \dots & \dots & \dots & \dots & \dots & \dots \\ 0 & 0 & 0 & \dots & 1 & -\lambda \end{vmatrix} = 0. \qquad (2)$$

При этом этот определитель легко раскрывается, и

$$P_n(\lambda) = (-1)^n(\lambda^n + p_1\,\lambda^{n-1} + p_2\,\lambda^{n-2} + \ldots + p_n). \qquad (3)$$

Используя какой-либо метод вычисления корней $P_n(\lambda)$ получаем метод вычисления собственных значений матрицы.

Подобный подход в настоящее время редко используется в силу следующей причины: корни полиномов (особенно высоких степеней) чувствительны к возмущению его коэффициентов. Наглядный пример привел Дж. Уилкинсон (1963 г.) Матрица $A = \mathrm{diag}(1, 2, 3, \ldots, 20)$ имеет хорошо отделенные с.з. $\lambda_k = k$, $1 \leqslant k \leqslant 20$, и его характеристический полином имеет целые коэффициенты:

$$P_{20}(\lambda) = \lambda^{20} - 210\,\lambda^{19} + p_2\,\lambda^{18} + \ldots + p_{20}.$$

Рассмотрим полином $\tilde{P}_{20}(\lambda)$, отличающийся от $P_{20}(\lambda)$ только коэффициентом перед λ^{19} равным $-210 + \varepsilon$, $\varepsilon = 2^{-23} \approx 10^{-7}$. Т.о. $\tilde{P}_{20}(\lambda) = P_{20}(\lambda) + \varepsilon\,\lambda^{19}$. Тщательные вычисления показывают, что лишь первые 10 с.з. полинома $\tilde{P}_{20}(\lambda)$ являются вещественными, а остальные с.з. имеют мнимые части порядка $O(1)$.

В силу указанной причины методы вычисления с.з. основаны на других идеях. Интересно отметить, что в MatLab функция *roots* вычисления корней полинома вида (3) основана на вычислении всех собственных значений матрицы определителя (2) при $\lambda = 0$ (сопровождающей матрицы полинома $P_n(\lambda)$).

2. Некоторые сведения из курса линейной алгебры. Напомним некоторые свойства с.з. и с.в. произвольных матриц.

1. Пусть σ_1, σ_2, …, σ_m — различные с.з. матрицы. Тогда для каждого σ_i найдется, по крайней мере, один соответствующий ему с.в. С.в., соответствующие различным с.з., линейно-независимы.

2. С.в. определяются не единственным образом: кратный с.в. также является с.в.; если z_1 и z_2 два линейно-независимых с.в., соответствующие λ, то их линейная комбинация $c_1 z_1 + c_2 z_2$ есть также с.в.,

соответствующий λ. Множество с.в., соответствующих с.з. λ, обозначается $U(\lambda)$ и называется собственным подпространством A, соответсвующим λ: $U(\lambda) = \{z \in \mathbb{R}^n : (A - \lambda I)\, z = 0\}$.

3. *Алгебраической кратностью* с.з. λ называется кратность λ как корня характеристического полинома. Максимальное число линейнонезависимых с.в., соответствующих с.з., называется его *геометрической кратностью*. Т.о. геометрическая кратность λ равна $\dim U(\lambda)$. Она не превосходит алгебраической кратности λ.

4. Матрицы A, A^{-1} и $B = \sum_{k=1}^{m} c_k\, A^k$ имеют одни и те же с.в., а с.з. матриц A^{-1} и B связаны с с.з. матрицы A равенствами

$$\lambda(A^{-1}) = \frac{1}{\lambda(A)}\,, \quad \lambda(B) = \sum_{k=1}^{m} c_k\, \lambda(A)^k.$$

Здесь и далее $\lambda(A)$ обозначает с.з. матрицы A.

Напомним следующие свойства с.з. и с.в. симметричных матриц.

1. С.з. симметричной матрицы являются вещественными числами, а с.в., соответствующие различным с.з. — ортогональными.

2. Если $A = A^T$, т.о. геометрическая кратность с.з. равна его алгебраической кратности. Как следствие, для каждого с.з. λ_i, $1 \leqslant i \leqslant n$, можно указать с.в. z_i так, что $\{z_i\}_{i=1}^n$ образуют ортонормированный базис в $\mathbb{R}^n$.

Следующие утверждения приведем без доказательств.

3. Пусть A, B две симметричные матрицы и задана некоторая нумерация с.з. $\lambda_i(A)$ матрицы A. Тогда найдется такая нумерация с.з. $\lambda_i(B)$ матрицы B, что

$$\max_{1 \leqslant i \leqslant n} |\lambda_i(A) - \lambda_i(B)| \leqslant \|A - B\|_2,$$
$$\sum_{i=1}^{n} (\lambda_i(A) - \lambda_i(B))^2 \leqslant \|A - B\|_E^2. \qquad (4)$$

Если $Ax = \lambda x$, $By = \mu y$, и μ отделено расстоянием γ от с.з. A, то

$$|\sin \angle(x, y)| \leqslant \|A - B\|_2 / \gamma.$$

Утверждение (4) известно как теорема Виландта — Хоффмана.

§ 1. Степенной метод и метод обратных итераций

Примем следующую нумерацию с.з. $\lambda_i = \lambda_i(A)$ матрицы A:

$$|\lambda_1| \leqslant |\lambda_2| \leqslant \cdots \leqslant |\lambda_{n-1}| \leqslant |\lambda_n|.$$

Через $z_1, \ldots, z_n$ будем обозначать соответствующие им *ортонормированные* с.в., через $\|\cdot\|$ и $(\cdot,\cdot)$ — евклидову норму и скалярное произведение.

1. Степенной метод. Степенной метод предназначен для отыскания собственного вектора, соответствующего максимальному по модулю с.з., т.е. z_n. Он определяется начальным приближением единичной длины x^0 к z_n, по которому образуется последовательность нормированных векторов $x^1, x^2, \ldots$ по правилу:

$$y^k = Ax^k, \quad x^{k+1} = y^k/\|y^k\|, \quad k = 0, 1, \ldots \tag{5}$$

а также последовательность приближений λ^k к с.з. λ_n:

$$\lambda^k = (Ax^k, x^k), \quad k = 1, 2, \ldots$$

Отметим, что название метода объясняется формулой $x^{k+1} = c_k\, A^k x^0$, которая следует из (5). Здесь c_k нормировочный множитель.

Для выяснения скорости сходимости метода, достаточно уметь оценивать убывание угла $\theta_k \in [-\pi/2, \pi/2]$ между векторами x^k и z_n, поскольку $\|x^k - z_n\| \leqslant |\theta_k|$. В самом, деле векторы x^k и z_n единичной длины, $\cos(\theta_k) = (x^k, z_n)$, $|\sin(\varphi)| \leqslant |\varphi|$ для любого φ, поэтому

$$\|x^k - z_n\|^2 = (x^k - z_n, x^k - z_n) = 2(1 - \cos(\theta_k)) = 4\,\sin^2(\theta_k/2) \leqslant \theta_k^2.$$

Теорема 1. *Пусть* $\rho = |\lambda_{n-1}|/|\lambda_n| < 1$, $(x^0, z_n) \neq 0$. *Тогда* $x^k \to z_n$, $\lambda^k \to \lambda_n$ *при* $k \to \infty$. *Более того,*

$$\|x^k - z_n\| \leqslant c\,\rho^k, \quad |\lambda^k - \lambda_n| \leqslant C\,\rho^{2k}, \tag{6}$$

где $c = |\operatorname{tg}\theta_0|$, $C = (|\lambda_{n-1}| + |\lambda_n|)\,\operatorname{tg}^2\theta_0$, θ_0 — *угол между* x^0 *и* z_n.

Доказательство. Рассмотрим двумерную плоскость, содержащую векторы z_n и x^k и пусть u^k есть единичный вектор этой плоскости, ортогональный к z_n. По мере продолжения итераций эта плоскость

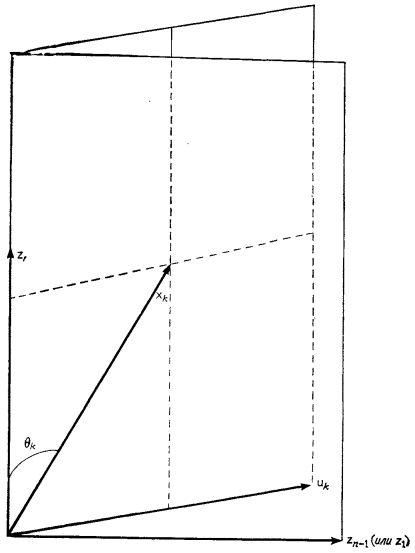

Рис. 1. Разложение x^k.

поворачивается вокруг фиксированной оси z_n (как раскрытое окно; см. рис. 1). По определению θ_k справедливо разложение

$$x^k = \cos\theta_k\, z_n + \sin\theta_k\, u^k. \tag{7}$$

В соответствии с изучаемым методом

$$x^{k+1} = c_k\, Ax^k = c_k \left(\cos\theta_k\, \lambda_n z_n + \|Au^k\|\, \sin\theta_k\, \frac{Au^k}{\|Au^k\|} \right), \tag{8}$$

где c_k — нормирующий множитель. Заметим, что единичный вектор $u^{k+1} = \frac{Au^k}{\|Au^k\|}$ ортогонален z_n, поскольку

$$(Au^k, z_n) = (u^k, Az_n) = \lambda_n (u^k, z_n) = 0.$$

Поэтому (8) есть разложение (7) на следующей $k+1$ итерации. Сравнивая эти формулы, получим

$$\cos\theta_{k+1} = c_k\, \cos\theta_k\, \lambda_n, \quad \sin\theta_{k+1} = c_k\, \sin\theta_k \|Au^k\|,$$

следовательно,

$$\operatorname{tg}\theta_{k+1} = \frac{\|Au^k\|}{\lambda_n}\, \operatorname{tg}\theta_k.$$

Разложим u^k по базису с.в. матрицы A и учтем ортогональность u^k и z_n. Получим, что $u^k = \sum\limits_{i=1}^{n-1} c_i z_i$, $\sum\limits_{i=1}^{n-1} c_i^2 = 1$, $Au^k = \sum\limits_{i=1}^{n-1} c_i \lambda_i z_i$, следовательно,

$$\|Au^k\|^2 = \sum_{i=1}^{n-1} c_i^2 \lambda_i^2 \leqslant \lambda_{n-1}^2 \sum_{i=1}^{n-1} c_i^2 = \lambda_{n-1}^2 \|u^k\|^2 = \lambda_{n-1}^2. \tag{9}$$

Таким образом,

$$|\operatorname{tg}\theta_{k+1}| \leqslant \rho\,|\operatorname{tg}\theta_k|, \quad k = 0, 1, \ldots \quad \Rightarrow \quad |\operatorname{tg}\theta_k| \leqslant \rho^k\,|\operatorname{tg}\theta_0|. \tag{10}$$

Это доказывает первую оценку в (6), т.к. $|\theta_k| \leqslant |\operatorname{tg}\theta_k|$.

Используем снова (7) и то, что $(Au^k, z_n) = 0$. Получим:

$$\begin{aligned}\lambda_n - \lambda^k &= \lambda_n - (A(\cos\theta_k\, z_n + \sin\theta_k\, u^k), \cos\theta_k\, z_n + \sin\theta_k\, u^k) = \\ &= \lambda_n - (\cos\theta_k\, \lambda_n z_n + \sin\theta_k\, Au^k, \cos\theta_k\, z_n + \sin\theta_k\, u^k) = \\ &= \lambda_n - (\lambda_n \cos^2\theta_k + (Au^k, u^k)\sin^2\theta_k) = (\lambda_n - (Au^k, u^k))\sin^2\theta_k,\end{aligned}$$

Откуда, вследствие (9), вытекает оценка

$$|\lambda_n - \lambda^k| \leqslant |\lambda_n + \lambda_{n-1}|\sin^2\theta_k \leqslant (|\lambda_n| + |\lambda_{n-1}|)\operatorname{tg}^2\theta_k. \tag{11}$$

Из (11) следует вторая оценка теоремы в силу (10). □

Замечание 1. 1. Условие $(x^0, z_n) \neq 0$ ($\theta_0 \neq \pm\pi/2$) на практике не слишком обременительно. Если оно нарушается, то при проведении итераций за счет ошибок округления приближения обязательно выйдут из гиперплоскости, ортогональной z_n.

2. Мы доказали сходимость метода в случае простого с.з. λ_n. Скорость сходимости метода тем выше (ρ тем меньше), чем лучше отделенность λ_n от остальных с.з. Если λ_n является кратным, например k кратным,

$$|\lambda_1| \leqslant \ldots \leqslant |\lambda_{n-k}| < |\lambda_{n-k+1}| = \ldots = |\lambda_n|,$$

то можно доказать, что справедлива оценка $|\lambda^k - \lambda_n| \leqslant C\,\rho^{2k}$, с $\rho = |\lambda_{n-k}/\lambda_n|$. Но сходимости x^k к какому-либо фиксированному вектору нельзя ожидать, т.к. в этом случае λ_n соответствует целое собственное подпространство с.в. $U(\lambda_n)$. Тем не менее, известно, что угол между векторами x^k и $U(\lambda_n)$ есть величина порядка $O(\rho^k)$.

3. Описанный метод без изменений применим и к несимметричным матрицам.

2. Метод обратных итераций. Метод предназначен для отыскания собственного вектора, соответствующего минимальному по модулю собственному значению и получается из степенного метода заменой A на A^{-1}. Опишем метод.

Выбираем нормированное начальное приближение x^0 и строим последовательность векторов $x^1, x^2, \ldots$ по формулам

$$y^k = A^{-1}x^k, \quad x^{k+1} = y^k/\|y^k\|,$$

а также числа $\lambda^k = (Ax^k, x^k)$, $k = 0, 1, 2, \ldots$.

При реализации метода выгоднее не строить и хранить матрицу A^{-1}, а решать на каждой итерации систему линейных уравнений $Ay^k = x^k$. Предварительно целесообразно представить матрицу A в факторизованном виде LL^T, LDL^T или QL разложения.

Теорема 2. *Пусть* $\rho = |\lambda_1|/|\lambda_2| < 1$, $(x^0, z_1) \neq 0$. *Тогда* $x^k \to z_1$, $\lambda^k \to \lambda_1$ *при* $k \to \infty$. *Более того,*

$$\|x^k - z_1\| \leqslant c\,\rho^k, \quad |\lambda^k - \lambda_1| \leqslant C\,\rho^{2k}, \tag{12}$$

где $c = |\operatorname{tg}\theta_0|$, $C = (|\lambda_1| + |\lambda_2|)\operatorname{tg}^2\theta_0$, θ_0 *есть угол между* x^0 *и* z_1.

Доказательство. Поскольку метод обратной итерации совпадает со степенным методом для матрицы $S = A^{-1}$, то надо найти максимальное по модулю с.з. ν_n матрицы S и соответственный ему с.в. Это будет собственная пара $(\nu_n = 1/\lambda_1, z_1)$. Следующее по величине с.з. равно $\nu_{n-1} = 1/\lambda_2$. Поэтому теорема непосредственно следуют из теоремы 1, с заменой λ_n, z_n и λ_{n-1} на $1/\lambda_1$, z_1 и $1/\lambda_2$, соответственно. □

3. Метод обратных итераций со сдвигом. Метод предназначен для отыскания собственного вектора, соответствующего собственному значению матрицы A, ближайшему к заданному числу σ (называемому сдвигом). Он получается из степенного метода заменой A на $S = (A - \sigma\, I)^{-1}$ и имеет следующий вид.

Выбираем нормированное начальное приближение x^0 и строим последовательность векторов $x^1, x^2, \ldots$ по формулам

$$y^k = (A - \sigma\, I)^{-1}x^k, \quad x^{k+1} = y^k/\|y^k\|,$$

а также числа $\lambda^k = (Ax^k, x^k)$, $k = 0, 1, 2, \ldots$.

Метод реализуется следующим образом. Перед началом итераций строится разложение $A - \sigma\, I = L_\sigma D_\sigma L_\sigma^T$ (LDL разложение). Далее на итерациях экономно решается система уравнений $(L_\sigma D_\sigma L_\sigma^T)y^k = x^k$ и получается x^k нормировкой вектора y^k.

Определим номер s собственного значения A, ближайшего к σ:

$$|\lambda_s(A) - \sigma| = \min_{i=1:n} |\lambda_i(A) - \sigma|.$$

Числу $\lambda_s(A)$ соответствует собственный вектор z_s. Тогда максимальное по модулю собственное значение S равно

$$|\lambda_n(S)| = \frac{1}{|\lambda_s(A) - \sigma|},$$

а числу $\lambda_n(S)$ также соответствует собственный вектор z_s. Поэтому метод обратных итераций со сдвигом σ действительно будет сходится к z_s. Как и для обратных итераций из теоремы 1 получается

Теорема 3. *Пусть* $\rho_s = |\lambda_s(A) - \sigma| / \min_{i \neq s} |\lambda_i(A) - \sigma| < 1$, $(x^0, z_s) \neq 0$. *Тогда* $x^k \to z_s$, $\lambda^k \to \lambda_s$ *при* $k \to \infty$. *Более того,*

$$\|x^k - z_s\| \leqslant c\,\rho_s^k, \quad |\lambda^k - \lambda_1| \leqslant C\,\rho_s^{2k}.$$

где постоянные c *и* C *не зависят от* k.

Отметим, что, если $\lambda_s(A)$ "хорошо" отделен от других собственных значений A, а σ достаточно близок к $\lambda_s(A)$, то ρ_s достаточно мал и итерации очень быстро сходятся. Это можно с пользой использовать при определении собственных векторов тогда, когда уже известно хорошее приближение к собственному числу.

§ 2. Метод вращений (Якоби)

Этот метод довольно часто используется при решении полной проблемы собственных значений для симметричных матриц не слишком высокого порядка. Напомним, что матрица $Q = Q_{kl} = (q_{ij})_{i,j=1}^n$, отличающаяся от единичной лишь четырьмя элементами: $q_{kk} = \cos\varphi$, $q_{ll} = \cos\varphi$, $q_{kl} = \sin\varphi$, $q_{lk} = -\sin\varphi$, где $1 \leqslant k < l \leqslant n$ — заданные целые числа, называется матрицей вращения. Она порождает преобразование поворота на угол φ в двумерной плоскости, натянутой на векторы канонического базиса с номерами k, l.

Опишем идею метода Якоби. Пусть A — симметричная матрица. Образуем матрицу T, столбцами которой являются ортонормированные с.в. $z_1, z_2, \ldots, z_n$ матрицы A. Нетрудно убедиться непосредственными вычислениями, что $T^T A T = \Lambda$, где Λ — диагональная матрица с элементами $\lambda_1, \lambda_2, \ldots, \lambda_n$ на диагонали. В методе вращений матрица T строится как предел последовательности ортогональных матриц T_s так, что $\lim\limits_{s\to\infty} T_s^T A T_s = \Lambda$, причем при каждом s матрица T_s конструируется как произведение матриц вращения.

Образуем по матрице A матрицу $\widehat{A} = Q^T A Q$, и попытаемся выбрать параметры матрицы вращения, т. е. k, l, φ, так, чтобы матрица $\widehat{A}$ была максимально близка к диагональной. Опуская элементарные выкладки, приведем выражение для суммы квадратов внедиагональных элементов матрицы $\widehat{A}$:

$$\sum_{i\neq j} \widehat{a}_{ij}^2 = \sum_{i\neq j} a_{ij}^2 - 2a_{kl}^2 + 2[a_{kl}\cos 2\varphi + \frac{1}{2}(a_{ll} - a_{kk})\sin 2\varphi]^2\,. \qquad (13)$$

Определим теперь числа k, l из условия:

$$|a_{kl}| = \max_{i \neq j} |a_{ij}| \tag{14}$$

а затем угол φ так, чтобы

$$a_{kl} \cos 2\varphi + \frac{1}{2}(a_{ll} - a_{kk}) \sin 2\varphi = 0,$$

или

$$\operatorname{tg} 2\varphi = \frac{2a_{kl}}{a_{kk} - a_{ll}}\,. \tag{15}$$

При указанном выборе параметров матрицы вращения сумма квадратов внедиагональных элементов матрицы $\widehat{A}$ принимает наименьшее значение. Это следует из формулы (13).

Лемма 1. *Пусть $\Sigma(A)$ есть сумма квадратов внедиагональных элементов матрицы A, параметры матрицы вращения Q определяются согласно формулам* (14), (15). *Тогда*

$$\Sigma(\widehat{A}) \leqslant q\,\Sigma(A), \quad q = 1 - \frac{2}{n(n-1)} \in (0,1). \tag{16}$$

Доказательство. Вследствие (15) имеем:

$$\Sigma(\widehat{A}) = \sum_{i \neq j} \widehat{a}_{ij}^2 = \Sigma(A) - 2a_{kl}^2, \tag{17}$$

а на основании (14)

$$\Sigma(A) \leqslant a_{kl}^2\, n(n-1). \tag{18}$$

Здесь учтено, что матрица порядка n имеет $n(n-1)$ внедиагональных элементов. Из (17), (18) очевидным образом следует (16). $\square$

Теперь можно описать метод Якоби. Пусть $A_0 = A$. Образуем последовательность матриц A_s, $s = 1, 2, \ldots$, с элементами $\{a_{ij}^{(s)}\}_{i,j=1}^{n}$, при помощи рекуррентной формулы:

$$A_{s+1} = Q_s^T A_s Q_s, \quad s = 0, 1, \ldots, \tag{19}$$

где параметры $k, l, \varphi^{(s)}$ матрицы вращения Q_s определяются так, что

$$|a_{kl}^{(s)}| = \max_{i \neq j} |a_{ij}^{(s)}|,$$

$$\operatorname{tg} 2\varphi^{(s)} = \frac{2a_{kl}^{(s)}}{a_{kk}^{(s)} - a_{ll}^{(s)}}\,.$$

Итерации проводят до тех пор, пока не будет выполнено условие

$$\Sigma(A_s) \leqslant \varepsilon^2 \tag{20}$$

для заданного ε. В этом случае в качестве приближений к с.з. матрицы A принимают диагональные элементы матрицы A_s, а столбцы матрицы $T_s = Q_0 Q_1 \dots Q_s$ считают приближениями к с.в. A.

Отметим, что из соотношений (19) и леммы 1 следует, что

$$A_{s+1} = T_s^T A T_s \quad T_s = Q_0 Q_1 \dots Q_s, \tag{21}$$

$$\Sigma(A_{s+1}) \leqslant q\,\Sigma(A_s) \quad \Rightarrow \quad \Sigma(A_s) \leqslant q^s\,\Sigma(A). \tag{22}$$

Отметим также, что T_s ортогональная матрица, как произведение ортогональных матриц, а из (22) следует, что условие (20) будет выполнено при некотором s.

При доказателстве сходимости это метода будем опираться на теорему Виланда — Хоффмана, из которой следует, что

$$|\lambda_k(A) - \lambda_k(B)| \leqslant \|A - B\|_E, \quad k = 1 : n. \tag{23}$$

Теорема 4. *Пусть s такое, что выполнено условие* (20). *Тогда найдется нумерация собственных значений матрицы A такая, что*

$$\max_{1 \leqslant k \leqslant n} |\lambda_k(A) - a_{kk}^{(s)}| \leqslant \varepsilon.$$

Доказательство. Пусть $\Lambda_s = \operatorname{diag}(a_{11}^{(s)}, a_{22}^{(s)}, \dots, a_{nn}^{(s)})$ и $B_s = A_s - \Lambda_s$. Имеем, $\lambda_i(\Lambda_s) = a_{ii}^{(s)}$. Из (21) следует, что матрицы A_s и A подобны, поэтому $\lambda_i(A_s) = \lambda_i(A)$.[1)] Поэтому в силу (23) получаем,

$$\begin{aligned} |\lambda_k(A) - a_{kk}^{(s)}|^2 &= |\lambda_k(A_s) - \lambda_k(\Lambda_s)|^2 \\ &\leqslant \|A_s - \Lambda_s\|_E^2 = \|B_s\|_E^2 = \Sigma(A_s) \leqslant \varepsilon^2, \quad k = 1 : n. \quad \square \end{aligned}$$

Согласно доказанной теореме, мы получаем собственные значения с гарантированной погрешностью, не превосходящей ε.

[1)] Матрицы A и B подобны, если найдется обратимая матрица U такая, что $B = U^{-1}AU$. У таких матриц с.з. совпадают: пусть $Ax = \lambda(A)x$ и $By = \lambda(B)y$. Тогда $U^{-1}AUy = \lambda(B)y$, т.е. $AUy = \lambda(B)Uy$ и $Ax = \lambda(B)x$ при $x = Uy$. Отсюда следует, что $\lambda(A) = \lambda(B)$.

§ 3. Понятие о QL-методе

Этот метод является одним из наиболее эффективных методов отыскания всех собственных значений симметричной матрицы не слишком высокого порядка. Метод состоит из двух основных шагов.

I). Методом отражений найдем разложение $T = UAU^T$, где U — ортогональная матрица, T — трехдиагональная матрица. Это можно сделать за $4/3n^3 + O(n^2)$ flop (см. далее теор. 5).

II). Начиная с матрицы $A_0 = T$ построим A_1, A_2, ... по следующему правилу для $s = 0, 1, \ldots$:

1) методом отражений строим QL разложение матрицы $A_s - \sigma_s I$:

$$A_s - \sigma_s I = Q_s L_s;$$

2) полагаем

$$A_{s+1} = L_s Q_s + \sigma_s I,$$

где сдвиги σ_s определяются по A_s по некоторому правилу.

Переход от A_s к A_{s+1} называется QL преобразованием матрицы. Нетрудно видеть, что

$$A_{s+1} = Q_s^T A_s Q_s,$$

где Q_s — ортогональная матрица. Поэтому с.з. A_s = с.з. T= с.з. A.

Эффективность метода объясняется следующими причинами:

i) все матрицы A_s оказываются симметричными (очевидно) и трехдиагональными (не очевидно, т.к. Q_s плотная матрица)!

ii) матрицу A_{s+1} по A_s можно вычислить за $O(n)$ flop.

iii) элемент $a_{21}^{(s)}$ матрицы A_s очень быстро стремиться к нулю с ростом s (почти всегда скорость сходимости лучше кубической), если в качестве σ_s выбирается с.з. матрицы

$$\begin{pmatrix} a_{11}^{(s)} & a_{12}^{(s)} \\ a_{21}^{(s)} & a_{22}^{(s)} \end{pmatrix},$$

расположенное ближе к $a_{11}^{(s)}$ (сдвиг по Уилкинсону).

Как только будет выполнено условие $a_{21}^{(s)} \approx 0$ с машинной точностью, то $a_{11}^{(s)}$ принимается за первое с.з. A.

Далее описанный выше шаг II) применяется к матрице $T = \bar{A}_s$ размера $n-1$, где $\bar{A}_s$ получается из A_s зачеркиванием первой строки и первого столбца и вычисляется второе с.з. A и т.д. На каждом шаге происходит понижение размерности матрицы на единицу. Аккуратная реализация этого метода позволяет найти все собственные значения T с машинной точностью за $\approx 9n^2$ flop (установлено экспериментально). Сравните это число с $4/3n^3 + O(n^2)$ flop на шаге I).

Если необходимо вычислить также все с.в. матрицы A, то первоначально найдем все с.в. матрицы T, последовательно используя метод обратных итераций со сдвигами, равными найденным с.з. При этом достаточно выполнить, как правило, лишь одну итерацию для определения каждого с.в. Найти все с.в. T можно за $O(n^2)$ flop. Зная их найдем и с.в. A (как? за сколько flop? опишите метод точнее!)

Теорема 5. *Пусть $A = A^T$. Тогда найдется ортогональная матрица U такая, что $T = UAU^T$, где T — трехдиагональная матрица.*

Доказательство. Представим матрицу A в блочном виде

$$A = \begin{bmatrix} \alpha_1 & a_1^T \\ a_1 & M_1 \end{bmatrix}.$$

Здесь α_1 — число, a_1 — столбец. Пусть

$$U_1 = \begin{bmatrix} 1 & 0^T \\ 0 & V_1 \end{bmatrix},$$

где V_1 — ортогональная матрица порядка $n-1$. Тогда

$$U_1 A U_1^T = \begin{bmatrix} \alpha_1 & (V_1 a_1)^T \\ V_1 a_1 & V_1 M_1 V_1^T \end{bmatrix}.$$

Рассуждая, как при доказательстве QR разложения матриц, мы можем построить матрицу V_1 так, чтобы все элементы столбца $V_1 a_1$, начиная со второго были равны нулю. Аналогичные рассуждения можно провести по отношению к матрице $V_1 M_1 V_1^T$ и так далее (см. описание QR метода). $\square$

Задания для самостоятельной работы

ВОПРОСЫ ДЛЯ САМОКОНТРОЛЯ

1. Дайте определение собственного значения и собственного вектора матрицы?

2. Сколько собственных значений существует у матрицы размера $n \times n$?

3. Сколько собственных векторов может соответствовать собственному значению матрицы размера $n \times n$? Приведите примеры.

4. Какой многочлен называется характеристическим многочленом матрицы? Какой смысл имеют его нули?

5. Что можно сказать о собственных векторах, соответствующих разным собственным м матрицы?

6. Что можно сказать о собственных значениях и векторах симметричных матрицы?

7. Пусть известны собственные значения и векторы матрицы A. Чему равны собственные значения и векторы матриц A^{-1} и A^2?

8. Сформулируйте теорему Виландта – Хоффмана. В чем смысл этой теоремы?

9. Приведите расчетные формулы степенного метода. Для решения какой задачи используется этот метод?

10. Приведите расчетные формулы метода обратных итераций. Для решения какой задачи используется этот метод?

11. Сформулируйте теорему о сходимости метода обратных итераций.

12. Приведите расчетные формулы метода обратных итераций со сдвигом. Для решения какой задачи используется этот метод?

13. Сформулируйте теорему о сходимости метода обратных итераций со сдвигом.

14. Для решения какой задачи предназначен метод вращений (Якоби)?

15. Дайте описание одной итерации метода вращений.

16. Сформулируйте теорему о сходимости метода вращений.

17. Позволяет ли метод вращений определить все собственные значения матрицы с точностью, скажем, 10^{-6}?

18. Какое преобразование матрицы называется QL преобразованием?

19. Дайте описание QL метода.

20. За сколько арифметических операций симметричную матрицу можно привести к трехдиагональному виду?

ЗАДАЧИ И УПРАЖНЕНИЯ

1. Докажите, что в степенном методе $x^{k+1} = A^{k+1}x^0/\|A^{k+1}x^0\|$.

2. Докажите, что если $A = A^T$, $Az = \lambda z$ и $u \perp z$, то $A^k u \perp z$.

3. Изучите сходимость степенного метода со сдвигом σ, когда A заменяется на $A - \sigma I$. Можно ли получить выгоду от введения сдвига?

4. Укажите реализацию метода, описанного в доказательстве теоремы 5, который требует порядка $4n^3/3$ flop.

5. Предполагая, что матрица Q в QL методе строится с использованием матриц отражения, докажите, что если матрица A — симметричная трехдиагональная матрица, то и все матрицы A_s, $s = 1, 2, \ldots$, также симметричны и трехдиагональны. Покажите, что QL разложение симметричной трехдиагональной матрицы требует $O(n)$ flop.

6. Выясните, какую структуру будет иметь матрица UAU^T, построенная в ходе доказательства теоремы 5, если отказаться от предположения об симметрии матрицы A. Матрицы такой структуры называются матрицами Хессенберга.

7. Пусть известно LDL разложение $A-\sigma I = L_\sigma D_\sigma L_\sigma^T$, где D_σ — диагональная матрица, и пусть n_σ — число отрицательных элементов на диагонали D_σ. а) Докажите, что n_σ равно числу собственных значений A, меньших чем σ; б) для решения каких задач можно использовать утверждение а)?

Указание. Используйте спектральное разложение A и совпадение инерции конгруэнтных матриц.

8. Сравните трудоемкость одной итерации степенного метода и метода обратных итераций со сдвигом для трехдиагональной матрицы.

9. Пусть z_n известно, $x^0 \perp z_n$, $\lambda_n(A)$ — простое собственное значение A. а) Докажите, что в степенном методе $x^k \to z_{n-1}$, $\lambda^k \to \lambda_{n-1}(A)$ при $k \to \infty$, если итерации осуществляется в точной арифметике. б) Как можно модифицировать степенной метод, чтобы найти z_{n-1} и $\lambda_{n-1}(A)$ при наличии округлений, если z_n известно?

ГЛАВА 4

Решение нелинейных уравнений

Нелинейные уравнения, как правило, не допускают решения в конечном виде, поэтому методы их решения являются приближенными и итерационными. Рассмотрения начнем со случая одного уравнения.

§ 1. Вычисление нулей функции одной переменной

Изучим классические методы решения нелинейного уравнения

$$f(x) = 0, \tag{1}$$

где f — заданная непрерывная функция вещественного переменного. Всюду в дальнейшем предполагается, что известен отрезок $[x_0, x_1]$, содержащий единственный корень α уравнения (1).

Определение 1. *Говорят, что последовательность* $\{x_k\}_{k=0}^{\infty}$ *сходится к корню*

a) с линейной скоростью (со скоростью геометрической прогрессии), если найдется $q < 1$ *такое, что*

$$|\alpha - x_{k+1}| \leqslant q\,|\alpha - x_k|, \quad k = 0, 1, \ldots$$

b) нелинейно, с порядком β*, если при некоторой постоянной* C

$$|\alpha - x_{k+1}| \leqslant C\,|\alpha - x_k|^{\beta}, \quad k = 0, 1, \ldots$$

При $\beta = 2$ говорят о квадратичной сходимости метода, при $\beta = 3$ — кубической и т.д. Ясно, что чем больше β, тем быстрее сходится ИМ.

1. Метод деления отрезка пополам. Будем считать, что $f(x_0)f(x_1) < 0$, т. е. функции f меняет знак в окрестности корня $\alpha \in (x_0, x_1)$. Положим $x_2 = (x_0 + x_1)/2$ и вычислим $f(x_0)f(x_2)$. Если $f(x_0)f(x_2) < 0$, то корень расположен на отрезке $[x_0, x_2]$. В противном случае — на отрезке $[x_2, x_1]$. Выбирая тот из двух отрезков, на котором лежит α, применяем к нему описанную процедуру деления

пополам и получим приближение x_3. Процесс продолжим до тех пор, пока длина отрезка не станет меньше заданного $\varepsilon > 0$. При этом корень, очевидно, будет найден также с точностью ε.

Нетрудно видеть, что итерации $\{x_k\}_{k=1}^{\infty}$ сходятся к корню линейно и $q = 1/2$. Метод пригоден для произвольной непрерывной функции f при условии $f(x_0)f(x_1) < 0$ и на каждой итерации требует лишь вычисления одного значения функции f.

2. Метод простой итерации Метод основан на приведении (1) к эквивалентному уравнению вида

$$x = \varphi(x) \tag{2}$$

и построении приближений $x_0, x_1, \ldots x_n, \ldots$ по формуле

$$x_{k+1} = \varphi(x_k), \quad k = 0, 1, 2, \ldots \tag{3}$$

Начальное приближение x_0 считается заданным.

Эквивалентное преобразование уравнения (1) к виду (2) может быть выполнено различными способами. Например, можно положить $\varphi(x) = x + \psi(x)f(x)$, где ψ — произвольная непрерывная функция, не обращающаяся в нуль в окрестности точки α. Функцию ψ следует выбрать так, чтобы обеспечить сходимость итераций.

Теорема 1. *Пусть α — решение (2) и в ее окрестности $S_r(\alpha) = \{x : |x - \alpha| \leqslant r\}$ функция φ удовлетворяет условию Липшица с постоянной меньшей единицы, т. е.*

$$|\varphi(x) - \varphi(y)| \leqslant q\,|x - y| \quad \forall\, x, y \in S_r(\alpha), \quad 0 < q < 1.$$

Тогда при любом $x_0 \in S_r(\alpha)$ метод сходится линейно и

$$|\alpha - x_k| \leqslant q^k\,|\alpha - x_0|. \tag{4}$$

Доказательство. Если $x_0 \in S_r(\alpha)$, то

$$|x_1 - \alpha| = |\varphi(x_0) - \alpha| = |\varphi(x_0) - \varphi(\alpha)| \leqslant q\,|x_0 - \alpha| < r,$$

следовательно, $x_1 \in S_r(\alpha)$. Аналогично, $x_1, x_2, \ldots \in S_r(\alpha)$. Т.о.

$$|x_k - \alpha| = |\varphi(x_{k-1}) - \varphi(\alpha)| \leqslant q\,|x_{k-1} - \alpha|\,, \quad k = 0, 1, \ldots,$$

откуда следует (4). □

Следствие 1. *Пусть φ непрерывно дифференцируема на $S_r(\alpha)$,*

$$|\varphi'(x)| \leqslant q < 1 \quad \forall\, x \in S_r(\alpha). \tag{5}$$

Тогда для x_k, построенной по методу (3)*, выполняется оценка* (4)*.*

Графическая интерпретация итерационного процесса (3) дана на рис. 1. Отметим, что случай $-1 < \varphi'(x) \leqslant 0$ наиболее интересен, так

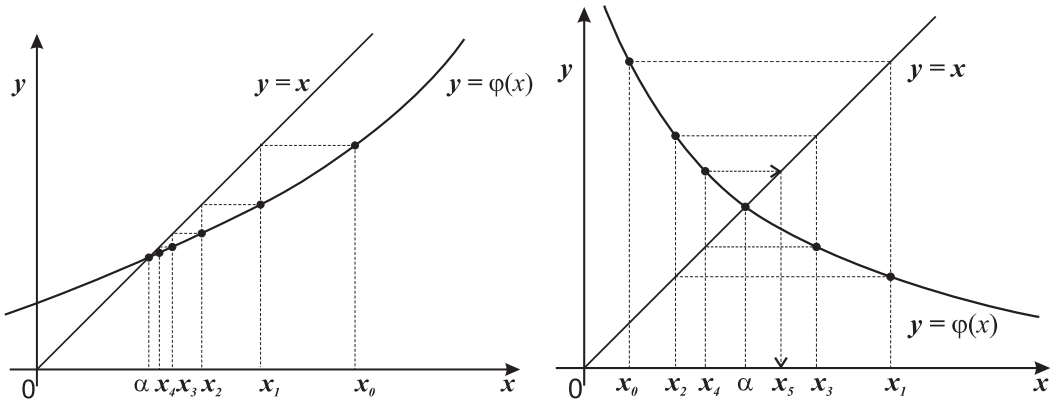

Рис. 1. К итерационному процессу (3). $0 \leqslant \varphi'(x) < 1$ (слева), $-1 < \varphi'(x) \leqslant 0$ (справа).

как при этом величина $|x_k - x_{k-1}|$ на каждом шаге итерационного процесса оценивает сверху погрешность приближения x_k к α.

Пример 1. Приведем пример определения функции φ. Пусть требуется вычислить $\sqrt{a}$, $a > 0$. Уравнение имеет вид

$$f(x) = x^2 - a = 0, \quad \alpha = \sqrt{a}. \tag{6}$$

Преобразуем это уравнение к виду

$$x = \varphi(x), \quad \text{где } \varphi(x) = \frac{a}{x}\,.$$

В этом случае $\varphi'(x) = -a/x^2$, $\varphi'(\sqrt{a}) = -1$, условие (5) в окрестности корня не выполнено, и итерационный процесс может разойтись, даже если x_0 лежит сколь угодно близко к $\sqrt{a}$ (проиллюстрируйте эту ситуацию графически!).

Перепишем теперь уравнение (6) в виде

$$x = \frac{1}{2}\left(x + \frac{a}{x}\right)$$

В этом случае

$$\varphi(x) = \frac{1}{2}\left(x + \frac{a}{x}\right), \quad \varphi'(x) = \frac{1}{2}\left(1 - \frac{a}{x^2}\right), \ \varphi'(\alpha) = 0$$

и, следовательно, $|\varphi'(x)| < 1$ в некоторой окрестности α. Таким образом, итерационный метод, определяемый как

$$x_{k+1} = \frac{1}{2}\left(x_k + \frac{a}{x_k}\right), \quad k = 0, 1, \ldots,$$

сходится при любом начальном приближении x_0, достаточно близком к $\sqrt{a}$. Более детальный анализ показывает, что сходимость имеет место при любом $x_0 > 0$ (убедитесь в этом!).

Теорема 2. *Пусть φ непрерывно дифференцируемо m раз и*

$$\varphi'(\alpha) = \varphi''(\alpha) = \ldots = \varphi^{(m-1)}(\alpha) = 0, \quad \varphi^{(m)}(\alpha) \neq 0, \quad m \geqslant 2.$$

Тогда метод простой итерации сходится нелинейно с порядком m, если x_0 выбрано достаточно близким к корню.

Доказательство. Поскольку $\varphi'(\alpha) = 0$, то найдется такая окрестность корня $S_r(\alpha)$, в которой $|\varphi'(x)| \leqslant q < 1$. Поэтому метод сходится, если $x_0 \in S_r(\alpha)$. Также все $x_k \in S_r(\alpha)$. Применяя формулу Тейлора для $\varphi(x_k)$ с остаточным членом в форме Лагранжа получим, что

$$|x_{k+1} - \alpha| = |\varphi(x_k) - \varphi(\alpha)| = \frac{1}{m!}\,|\varphi^{(m)}(\xi_k)|\,|x_k - \alpha|^m \leqslant C\,|x_k - \alpha|^m,$$

где C есть максимум $|\varphi^{(m)}(x)|/m!$ в $S_r(\alpha)$. $\square$

3. Методы определения φ. Рассмотрим некоторые общие способы построения функции φ.

1) Метод Ньютона. В этом случае полагают

$$\varphi(x) = x - \frac{f(x)}{f'(x)},$$

т. е.

$$x_{k+1} = x_k - \frac{f(x_k)}{f'(x_k)}, \ k = 0, 1, \ldots \tag{7}$$

Нетрудно видеть, что

$$\varphi'(\alpha) = 0, \quad \varphi''(\alpha) \neq 0, \tag{8}$$

если $f(\alpha) = 0$, а $f'(\alpha) \neq 0$, т. е. α — простой корень уравнения (1). Отсюда вытекает, что метод Ньютона (7) сходится при любом начальном приближении x_0, достаточно близком к α.

Равенства (8) показывают, что метод Ньютона — метод второго порядка и, следовательно, имеет квадратичную сходимость.

Метод Ньютона имеет простую геометрическую интерпретацию: x_{k+1} — точка пересечения касательной к графику функции f в точке $(x_k, f(x_k))$ с осью x (см. левый рис. 2).

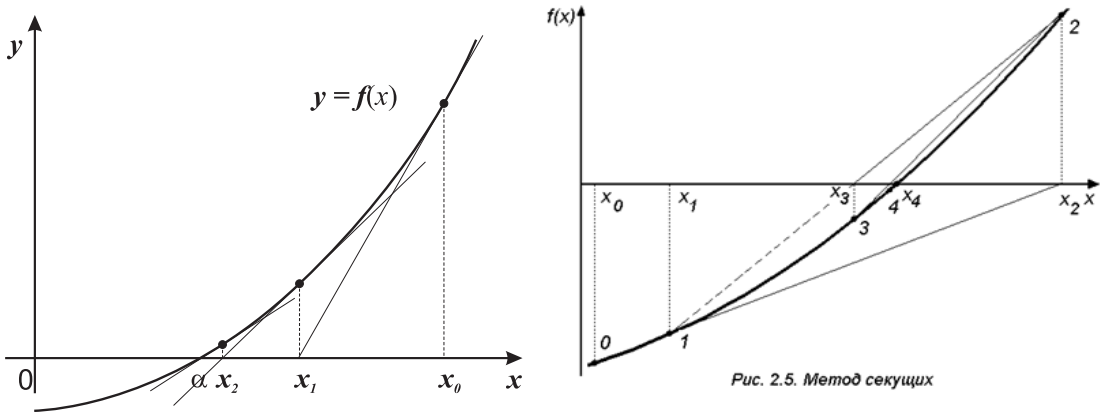

Рис. 2. Иллюстрация метода Ньютона (рис. слева) и метода секущих (рис. справа).

2) Метод секущих. Недостатком метода Ньютона является необходимость вычисления производной функции f. В некоторых случаях такая информация о функции может быть недоступна. Метод секущих получается при аппроксимации производной $f'(x_k)$ в методе Ньютона (7) разностным отношением

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

В результате приходим к следующему итерационному методу:

$$x_{k+1} = x_k - \frac{(x_k - x_{k-1})f(x_k)}{f(x_k) - f(x_{k-1})} \tag{9}$$

(см. правый рис. 2). Геометрически x_{k+1} — точка пересечения секущей, проходящей через точки $(x_k, f(x_k))$ и $(x_{k-1}, f(x_{k-1}))$ с осью x.

В отличие от ранее рассмотренных методов для вычисления нового приближения x_{k+1} требуется использовать два предыдущих приближения к корню. Такие методы называются двухшаговыми.

Вновь используя разложение по формуле Тейлора, можно показать, что справедливо приближенное равенство

$$x_{k+1} - \alpha \approx \frac{1}{2}\,\frac{f''(\alpha)}{f'(\alpha)}\,(x_{k-1} - \alpha)(x_k - \alpha)\,, \tag{10}$$

показывающее, что если имеет место сходимость метода (9), то его погрешность убывает несколько медленнее, чем в методе Ньютона.

Из (10) выводится, что метод секущих сходится нелинейно с порядком $\beta = (1 + \sqrt{5})/2 \approx 1.6$. Если учесть, что на каждой итерации этого метода требуется лишь одно вычисление функции f (второе берется с предыдущей итерации), то он по эффективности превосходит метод Ньютона.

3) Метод хорд. Если известен интервал (x_0, x_1), на концах которого функция f имеет противоположные знаки, то приближение x_2 к корню можно определить как точку пересечения прямой, проходящей через точки $(x_0, f(x_0))$ и $(x_1, f(x_1)$ с осью x (см. рис. 3). Для x_2

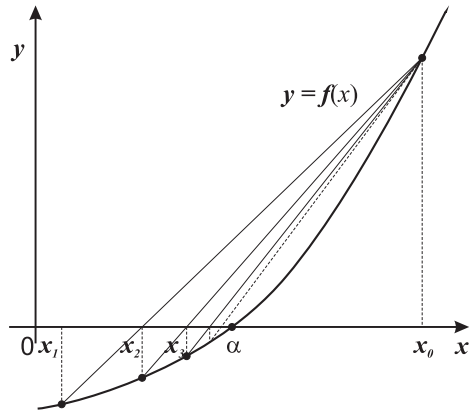

Рис. 3. Метод хорд.

получается следующая формула:

$$x_2 = \frac{x_0 f(x_1) - x_1 f(x_0)}{f(x_1) - f(x_0)}\,.$$

Все последующие приближения будем вычислять по формуле

$$x_{k+1} = \frac{x_0 f(x_k) - x_k f(x_0)}{f(x_k) - f(x_0)}\,, \quad k = 2, 3, \ldots \tag{11}$$

Метод (11) можно переписать в виде (3), где

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

Используя формулу Тейлора, нетрудно, получить, что

$$\varphi'(\alpha) = \frac{f''(\xi)}{f'(\alpha)}\,(x_0 - \alpha)^2, \quad \xi \in (x_0, x_1).$$

Отсюда вытекает, что $|\varphi'(\alpha)| < 1$, если x_0 выбрано достаточно близко к корню, т.е. существует окрестность α, в которой $|\varphi'(x)| \leqslant q < 1$, и метод (11) сходится при любом x_1 из этой окрестности линейно.

§ 2. Методы решения систем нелинейных уравнений

Рассмотрим задачу решения системы нелинейных уравнений

$$\begin{cases} f_1(x_1, \ldots, x_n) = 0, \\ f_2(x_1, \ldots, x_n) = 0, \\ \ldots\ldots\ldots\ldots\ldots \\ f_n(x_1, \ldots, x_n) = 0, \end{cases} \tag{12}$$

где $f_1, f_2, \ldots, f_n$ — заданные функции n вещественных переменных. Как и в случае одного уравнения, будем предполагать, что система (12) имеет корень $\alpha = (\alpha_1, \alpha_2, \ldots, \alpha_n)$.

Систему (12) хапишем в векторном виде

$$F(x) = 0, \tag{13}$$

где $F(x) = (f_1(x), f_2(x), \ldots, f_n(x))^T : \mathbb{R}^n \to \mathbb{R}^n$ — вектор-функция.

1. Метод простой итерации. В этом методе систему (13) предварительно некоторыми эквивалентными преобразованиями приводят к виду

$$x = G(x) \tag{14}$$

и, начиная с некоторого начального приближения $x^0 = (x_1^0, x_2^0, \ldots, x_n^0)$, строят последовательность векторов $x^k = (x_1^k, x_2^k, \ldots, x_n^k)$ по следующему правилу

$$x^{k+1} = G(x^k), \quad k = 0, 1, 2, \ldots \tag{15}$$

Существуют различные способы приведения системы (13) к эквивалентной системе вида (14). Обычно, они естественным образом подсказываются структурой системы (12).

Часто поступают следующим образом. Выбирают некоторую невырожденную матрицу B и параметр $\tau > 0$ и полагают $G(x) = x + \tau B^{-1}F(x)$. Тогда x^k, согласно (15), определяются как

$$x^{k+1} = x^k + \tau B^{-1}F(x^k), \quad k = 0, 1, 2, \ldots$$

или, эквивалентно,

$$B\,\frac{x^{k+1} - x^k}{\tau} + F(x^k) = 0, \quad k = 0, 1, 2, \ldots$$

Выбор матрицы B и τ диктуется условиями сходимости итераций.

2. Метод Ньютона. При наличии достаточно хорошего приближения к решению системы (13) очень часто весьма эффективным оказывается его уточнение по методу Ньютона. Этот метод является непосредственным обобщением метода Ньютона для одного нелинейного уравнения и может быть записан следующим образом

$$x^{k+1} = x^k - (F'(x^k))^{-1}F(x^k), \quad k = 0, 1, 2, \ldots \tag{16}$$

Здесь

$$F'(x^k) = \left\{ \frac{\partial f_i(x^k)}{\partial x_j} \right\}_{i,j=1}^{n}$$

есть матрица Якоби отображения F, вычисленная в точке x^k. При практической реализации метода его обычно переписывают в виде

$$F'(x^k)(x^{k+1} - x^k) = -F(x^k),$$

или

$$F'(x^k)\Delta^k = -F(x^k), \quad x^{k+1} = x^k + \Delta^k.$$

Таким образом, для построения x^{k+1} по известному x^k нужно решить систему линейных алгебраических уравнений с матрицей $F'(x^k)$.

При начальном приближении, достаточно близком к корню, он сходится квадратично, т.е.

$$\|x^{k+1} - \alpha\| \leqslant \text{const}\,\|x^k - \alpha\|^2.$$

Как правило, формирование матрицы Якоби $F'(x^k)$ оказывается существенно более трудоемкой задачей, чем вычисление вектора $F(x^k)$. Поэтому часто используются различные модификации метода Ньютона. Отметим следующие два.

1) Используем вместо матрицы Якоби ее аппроксимацию разделенными разностями:

$$F'(x) = \left\{ \frac{\partial f_i(x^k)}{\partial x_j} \right\}_{i,j=1}^{n} \approx B_k = \left\{ \frac{F(x + h\,e_j) - F(x)}{h} \right\}_{j=1}^{n},$$

где $e_j = (0, \ldots, 1, 0, \ldots, 0)$ — j-тый координатный вектор, h — достаточно малое число.[1)] В итоге придем к методу секущих, который имеет следующий вид:

$$B_k \Delta^k = -F(x^k), \quad x^{k+1} = x^k + \Delta^k.$$

При начальном приближении, достаточно близком к корню, он сходится нелинейно, причем

$$\|x^{k+1} - \alpha\| \leqslant \text{const}\, \|x^k - \alpha\|^\beta, \quad \beta = (1 + \sqrt{5})/2 \approx 1.6.$$

2) Рассмотрим итерации

$$F'(x^0) \Delta^k = -F(x^k), \quad x^{k+1} = x^k + \Delta^k.$$

Здесь матрица Якоби вычисляется только на начальном приближении. При этом экономия достигается как sa счет формирования матрицы системы, так и за счет того, что решение нескольких систем с одинаковыми матрицами существенно проще, чем решение того же количества систем с разными матрицами. Через определенное число итераций матрицу можно перевычислять.

При этом, однако, надо иметь в виду, что если метод (16) обладает при достаточно хорошем начальном приближении квадратичной сходимостью, то модифицированный метод Ньютона сходится лишь линейно.

3. Вариационные методы. Большой класс методов решения системы (12) основан на ее эквивалентной формулировке в виде задачи минимизации функции n переменных:

$$\Phi(x) = \min_{y \in \mathbb{R}^n} \Phi(y), \quad \Phi(y) = f_1^2(y) + f_2^2(y) + \ldots + f_n^2(y).$$

Изучению методов такого типа посвящена учебная дисциплина “Методы оптимизации“.

[1)] Как видим, трудоемкость вычисления B_k равна трудоемкости $n + 1$ вычисления функции F.

Задания для самостоятельной работы

ВОПРОСЫ ДЛЯ САМОКОНТРОЛЯ

1. Что понимается под линейной сходимостью последовательности $\{x^k\}_{k=0}^{\infty}$ к корню α?

2. Что понимается под нелинейной (квадратичной) сходимостью последовательности $\{x^k\}_{k=0}^{\infty}$ к корню α?

3. а) Опишите метод деления отрезка пополам. б) для каких функций он применим; с) с какой скоростью он сходится?

4. В чем заключается метод простых итераций для решения уравнения $f(x) = 0$?

5. Сформулируйте теорему о линейной сходимости метода простых итераций.

6. Сформулируйте теорему о нелинейной сходимости метода простых итераций.

7. Приведите примеры итерационных функций $\varphi(x)$.

8. Опишите метод Ньютона. Дайте его геометрическую интерпретацию.

9. Опишите метод секущих. Дайте его геометрическую интерпретацию.

10. Опишите метод хорд. Дайте его геометрическую интерпретацию.

11. Сформулируйте задачу решения системы нелинейных уравнений. Всегда ли такая система имеет решение?

12. Приведите формулы метода Ньютона для решения системы нелинейных уравнений.

13. Каким образом задачу решения системы нелинейных уравнений можно свести к задаче на безусловный минимум функции многих переменных?

ЗАДАЧИ И УПРАЖНЕНИЯ

1. а) Получите формулы метода Ньютона для решения уравнения $x^2 = a$. б) При каких начальных приближениях он сходится? с) Как быстро? d) используя компьютер (или калькулятор) найдите $\sqrt{2}$ с 6 знаками после запятой начиная с $x^0 = 1.4$.

2. Укажите пример функции, для которой метод Ньютона: а) сходится при любом начальном приближении; б) сходится при начальном приближении, достаточно близком к корню, иначе расходится.

3. Напишите две функции, реализующие метод Ньютона и секущих на выбранном языке прораммирования. Устройте соревнование этих функций при решении тестовых уравнений (с известными решениями). Наблюдайте сходимость и число итераций. Обоснованно определите лучший метод.

4. Двумя разными способами сведите задачу $e^{-x} - \sin(x) = 0$ к виду $x = \varphi(x)$ так, чтобы методом простой итерации $x^{k+1} = \varphi(x^k)$ найти корень $\alpha \approx 0.5885$. а) Обоснуйте выбор φ. б) реализуйте итерации и проверьте результат практически.

5. Пусть $g(x) = \sqrt{1 + x^2}$. Покажите, что метод Ньютона для уравнения $g'(x) = 0$ имеет следующие свойства: а) если $|x^0| < 1$, то x^k монотонно стремятся к нулю при $k \to \infty$. б) если $|x^0| > 1$, то $|x^k|$ монотонно стремятся к ∞ при $k \to \infty$.

6. Пусть α есть корень $f(x)$ кратности 2. С какой скоростью сходится метод Ньютона в этом случае, если начальное приближение выбрано достаточно близко к корню?

ГЛАВА 5

Методы решения задачи Коши для систем ОДУ

К обыкновенным дифференциальным уравнениям (ОДУ) сводится изучение многообразных задач в математике и в различных предметных областях (в физике, химии, биологии, медицине и т.д.).

Конкретная задача может приводить к дифференциальному уравнению любого порядка, или к системе уравнений различных порядков. Но известно, что уравнение n-го порядка

$$u^{(n)}(x) = f(x, u'(x), u''(x), \ldots, u^{(n-1)}(x))$$

при введении новых неизвестных $u_i(x) = u^{(i)}(x)$ можно свести к эквивалентной системе n уравнений первого порядка

$$\begin{aligned} u_i'(x) &= u_{i+1}, \quad i = 0, 1 : n-2, \\ u_{n-1}'(x) &= f(x, u_0, u_1, \ldots, u_{n-1}), \end{aligned}$$

где $u_0(x) = u(x)$. Аналогично, произвольную систему ОДУ любого порядка можно заменить эквивалентной системой первого порядка

$$u_i'(x) = f_i(x, u_1(x), \ldots, u_n(x)), \quad i = 1 : n, \tag{1}$$

записывая их для краткости в векторной форме

$$u'(x) = f(x, u(x)), \tag{2}$$

где u и f вектор-функции: $u(x) = (u_1(x), \ldots, u_n(x))^T(x)$,

$$f(x, u(x)) = (f_1(x, u(x)), \ldots, f_n(x, u(x)))^T.$$

Общее решение системы (1) зависит от n параметров $c_1, c_2, \ldots, c_n$ — постоянных интегрирования. Для выделения единственного решения необходимо наложить n дополнительных условий на функции $u_i(x)$.

В зависимости от этих условий различают *задачи Коши* и *краевые задачи*. В случае задачи Коши дополнительные условия имеют вид

$$u_i(a) = u_{ai}, \quad i = 1 : n, \tag{3}$$

т. е. заданы значения всех функций u_i в одной и той же точке $x = a$. В случае краевой задачи одна часть дополнительных условий задается в точке $x = a$, а другая — в точке $x = b$.

Решение задачи (1), (3), как правило, требуется найти на некотором отрезке $a \leqslant x \leqslant b$. Условия (3) в векторной записи имеют вид $u(a) = u_a$.

Редкие ОДУ, а тем более их системы, могут быть решены точно. Например, доказано, что решение первого из уравнений

$$u'(x) = x^2 + u^2(x), \quad u'(x) = \frac{u(x) - x}{u(x) + x}$$

не выражается через элементарные функции, а первый интеграл второго уравнения, имеющий вид

$$0.5 \ln(x^2 + u^2) + \operatorname{arctg}(u/x) = c$$

неявно определяет $u(x)$: для вычисления $u(x)$ надо решить это трансцендентное уравнение при заданном x. Это можно сделать только приближенно, что не проще, чем приближенно решить само ОДУ.

Далее мы изучим ряд приближенных методов решения задачи Коши. Мы ограничимся рассмотрением одного уравнения первого порядка. Векторная запись задачи Коши позволяет без изменения вида формул обобщить рассматриваемые методы на системы уравнений.

1. О методах решения задачи Коши. Для заданной функции $f(x, p)$ и $u_a \in \mathbb{R}$ рассматривается задача Коши

$$u'(x) = f(x, u(x)), \quad a < x \leqslant b, \tag{4}$$

$$u(a) = u_a. \tag{5}$$

Большинство из применяемых в настоящее время приближенных методов решения этой задачи основано на том, что отыскивается приближение к решению $u(x)$ на некотором заданном или генерируемом в ходе решения задачи множестве точек $x_0, x_1, \ldots, x_N$, называемом сеткой узлов. Такие методы, обычно, называются сеточными.

Если $x_i - x_{i-1} = h$, $i = 1, \ldots, N$, то сетку называют равномерной с шагом h. Через $u_i = u(x_i)$, $y_i = y(x_i)$ будем обозначать соответственно точное и приближенное решение задачи в точке сетки с номером

i. В отличие от $u(x)$ функции $y(x)$ определена только в точках сетки (является сеточной функцией). При необходимости (например, с целью графического представления) она может быть продолжена на весь отрезок $[a, b]$, скажем, при помощи сплайн интерполяции.

В дальнейшем, для упрощения изложения, сетка предполагается равномерной. Там где переход к неравномерной сетке принципиален, будут сделаны специальные замечания.

2. Метод разложения в ряд Тейлора. В точке сетки $x_0 = a$ решение известно, поэтому положим $y_0 = u_a$. Если функция $f(x, p)$ обладает достаточным числом производных, то по значению $u(x)$ в точке $x_0 = a$ можно сколь угодно точно вычислить решение в точке $x_1 = x_0 + h$. Это важная особенность задачи Коши для ОДУ.

Запишем $u(x_1) = u(x_0 + h)$ по формуле Тейлора:

$$u(x_1) = u(x_0) + h\,u(x_0) + \frac{h^2}{2!}\,u''(x_0) + \cdots + \frac{h^n}{n!}\,u^{(n)}(x_0) + \frac{h^{n+1}u^{(n+1)}(\xi)}{(n+1)!}\,,$$

где $\xi \in [x_0, x_1]$. Отбрасывая остаточный член, который предполагается малым, приближенное решение определим по формуле

$$y(x_1) = u(x_0) + h\,u'(x_0) + \frac{h^2}{2!}\,u''(x_0) + \cdots + \frac{h^n}{n!}\,u^{(n)}(x_0). \qquad (6)$$

Чем больше членов суммы здесь удастся записать, тем точнее приближает $y(x_1)$ значение $u(x_1)$. При этом справедлива оценка точности

$$|u(x_1) - y(x_1)| \leqslant M_{n+1}\,\frac{h^{n+1}}{(n+1)!}\,, \quad M_{n+1} = \max_{x\in[a,b]} |u^{(n+1)}(x)|. \qquad (7)$$

Значения производных $u^{(k)}(x_0)$ в (6) можно вычислить, так как $u(x_0) = u_a = y_0$ известно. В самом деле, т.к. $u(x)$ — решение (4), то

$$u'(x_0) = f(x_0, u(x_0)) = f(x_0, y_0).$$

Дифференцируя (4), получим

$$u''(x) = \left(f\left(x, u(x)\right)\right)' = \frac{\partial f}{\partial x} + \frac{\partial f}{\partial u}\,u' = \frac{\partial f}{\partial x} + \frac{\partial f}{\partial u}\,f(x, u)\,. \qquad (8)$$

Следовательно,

$$u''(x_0) = \frac{\partial f(x_0, y_0)}{\partial x} + \frac{\partial f(x_0, y_0)}{\partial u}\,f(x_0, y_0)\,.$$

Дифференцируя (8), аналогично найдем $u'''(x_0)$ и т.д. Вычислив $y(x_1)$, можно определить $y(x_2)$ по указанной выше схеме вычислений (заменяя в формулах (x_0, y_0) на (x_1, y_1)). Таким образом можно построить значения $y(x_i)$ во всех точках сетки.

Как видно, формулы для производных с увеличением их порядка становятся все более и более громоздкими. Для вычисления производных можно использовать пакеты программ аналитических вычислений (например, MAPLE, MATLAB).

Есть области приложений, где метод разложения в ряд Тейлора издавна используется. Например, это некоторые задачи небесной механики, где приходится многократно интегрировать одни и те же дифференциальные уравнения движения при различных значениях параметров.

3. Об одношаговых и многошаговых методах. Универсальные программы, рассчитанные на широкие классы дифференциальных уравнений, основываются на таких приближенных методах, которые вообще не используют производных функции $f(u, p)$. В пользу применения именно таких методов говорит и то, что довольно часто возникают прикладные задачи, в которых функция $f(u, p)$ не задается аналитически, а значения ее вырабатываются некоторым вычислительным алгоритмом или получаются в результате измерений (физических параметров некоторого процесса).

Указанные методы довольно четко разделяются на два больших класса — одношаговые и многошаговые методы. Примерами первых являются методы типа Рунге — Кутты, примерами вторых — методы типа Адамса, котрые мы изучим далее.

С общей точки зрения эти методы описываются очень просто. Любой одношаговый метод может быть определен так:

$$y_{i+1} = \Phi(y_i), \quad i = 0, 1, \ldots, N-1, \quad y_0 = u_a, \tag{9}$$

где под Φ понимается способ вычисления решения в следующей точке сетки x_{i+1} по известному значению решения y_i в предыдущей точке сетки. Отметим, что метод разложения в ряд Тейлора, рассмотренный выше, принадлежит к таким методам.

Общая формула для многошаговых методов такова

$$y_{i+1} = \Phi(y_i, y_{i-1}, y_{i-2}, \ldots, y_{i-k}), \; i = k, \ldots, N-1, \quad y_0 = u_a, \tag{10}$$

где Φ вновь — некоторая функция, но на этот раз зависящая от значений приближенного решения в $k+1$ точках. Число k — параметр метода, определяющий его точность. Для начала счета по этим формулам необходимо знать кроме y_0 также $y_1, y_2, \ldots, y_k$, которые должны быть вычислены каким-либо другим методом с необходимой точностью.

4. Простейшие сеточные методы. Пусть $u(x)$ есть решение задачи Коши. Проинтегрируем уравнение $u' = f(x,u)$ по отрезку $[x_i, x_{i+1}]$ и результат поделим на h. Получим

$$\frac{u(x_{i+1}) - u(x_i)}{h} = \frac{1}{h}\int_{x_i}^{x_{i+1}} f(x,u(x))\,dx. \tag{11}$$

Дальнейшее, фактически, основано на том или ином способе приближенного вычисления участвующего здесь интеграла.

4.1. Явный метод Эйлера. Используем сначала простую формулу — квадратуру левых прямоугольников ($\phi(x) = f(x,u(x)) = u'(x)$):

$$\int_{x_i}^{x_{i+1}} \phi(x)\,dx = h\,\phi(x_i) + \mu_{i+1}, \quad \mu_{i+1} = \frac{h^2}{2}\,\phi'(\xi_i).$$

Здесь μ_{i+1} — погрешность квадратуры, $\xi_i \in [x_i, x_{i+1}]$. Получим

$$\frac{u(x_{i+1}) - u(x_i)}{h} = f(x_i, u(x_i)) + \psi_{i+1}, \quad \psi_{i+1} = \frac{h}{2}\,u''(\xi_i).$$

Считая h малой величиной, отбросим здесь неизвестное слагаемое ψ_{i+1} порядка $O(h)$. Заменяя $u(x)$ на $y(x)$, получим равенство

$$\frac{y(x_{i+1}) - y(x_i)}{h} = f(x_i, y(x_i)).$$

Эти рассуждения справедливы для всех $i = 0 : N-1$. Т.о. получаем

$$y_{i+1} = y_i + hf(x_i, y_i), \quad i = 0 : N-1, \quad y_0 = u_a.$$

Эти формулы определяют явный метод Эйлера (метод Рунге — Кутты первого порядка точности).

Величина ψ_i называется погрешностью аппроксимации. Как увидим далее, она определяет порядок точности метода, т.к. справедлива оценка (см. далее)

$$\|u - y\|_\infty = \max_{1\leqslant i\leqslant N} |u(x_i) - y_i| \leqslant C \max_{1\leqslant i\leqslant N} |\psi_i|\,,$$

где C — некоторая постоянная, не зависящая от h. Из этой оценки следует, что $\|u-y\|_\infty = O(h)$, т.е. метод Эйлера не может обеспечить высокой точности, если шаг сетки не слишком мал. Тем не менее на практике он используется довольно часто.

4.2. Неявный метод Эйлера. Используем для приближенного вычисления интеграла формулу правых прямоугольников

$$\int_{x_i}^{x_{i+1}} \phi(x)\,dx = h\,\phi(x_{i+1}) + \mu_{i+1}, \quad \mu_{i+1} = -\frac{h^2}{2}\,\phi'(\xi_i).$$

Рассуждая как и ранее, приходим к следующему методу:

$$y_{i+1} = y_i + hf(x_{i+1}, y_{i+1}),\ i = 0, 1, \ldots, N-1,\ y_0 = u_a.$$

Он называется неявным методом Эйлера, т.к. y_{i+1} не определяется явно по известному y_i, а разыскивается как решение уравнения

$$y = g(y), \quad g(y) = y_i + hf(x_{i+1}, y). \tag{12}$$

Его решение, если функция $f(x,p)$ не слишком простая, не может быть найдено в явном виде. Приходится применять приближенные (итерационные) методы, например, метод простой итерации. Согласно ему, выбирается некоторое начальное приближение $y_{i+1}^{(0)}$ к y_{i+1}. Обычно, берут $y_{i+1}^{(0)} = y_i$ или определяют $y_{i+1}^{(0)}$ при помощи явного метода Эйлера по известному y_i. Затем строят последовательность приближений

$$y_{i+1}^{(k+1)} = y_i + hf(x_{i+1}, y_{i+1}^{(k)}), \quad k = 0, 1, \ldots$$

Этот итерационный метод обычно быстро сходится, что объясняется малостью h ($g'(y) = hf'_y(x_{i+1}, y)$ и $|g'(y)| = O(h)$). На практике редко выполняют больше двух-трех итераций. Более того, за сходимостью метода не следят, а назначают априори, волевым решением, некое число итераций, например, две. Тогда метод полностью определен.

Естественный вопрос: зачем нужен неявный метод, если по точности он, скорее всего, не лучше явного, поскольку порядок точности формул левых и правых прямоугольников одинаковы? Ответ: явный и неявный методы существенно различаются в отношении такого важного свойства, как устойчивость метода (об этом речь впереди).

4.3. Метод центральных прямоугольников. Увеличим точность вычисления интеграла (11):

$$\int_{x_i}^{x_{i+1}} \phi(x)\,dx = h\,\phi(x_i + 0.5\,h) + \mu_{i+1}, \quad \mu_{i+1} = \frac{h^3}{24}\,\phi''(\xi_i),$$

т.е. используем квадратуру центральных прямоугольников. Получим

$$\frac{u(x_{i+1}) - u(x_i)}{h} = f(x_i, u(x_i + 0.5\,h)) + \psi_{i+1}, \quad \psi_{i+1} = \frac{h^2}{24}\,u'''(\xi_i).$$

Значение $u(x_i+0.5\,h)$ определим явным методом Эйлера с шагом $h/2$:

$$u(x_i + 0.5\,h) = u(x_i) + \frac{h}{2}\,f(x_i, u_i) + \mu_{i+1/2}, \quad \mu_{i+1/2} = \frac{h^2}{8}\,u''(\overline{\xi}_i).$$

Отбрасывая погрешности ψ_{i+1}, $\mu_{i+1/2}$ порядка $O(h^2)$, получим формулы (пример метода Рунге — Кутты второго порядка точности):

$$y_{i+1} = y_i + h\,f(x_i + h/2, y_i + h/2\,f(x_i, y_i)), \quad i = 0 : N-1, \quad y_0 = u_a.$$

Все вычисления здесь проводятся по явным формулам.

4.4. Метод предиктор-корректор. Используем для приближенного вычисления интеграла формулу трапеций:

$$\int_{x_i}^{x_{i+1}} \phi(x)\,dx = \frac{h}{2}\,(\phi(x_i) + \phi(x_{i+1}) + \mu_{i+1}, \quad \mu_{i+1} = -\frac{h^3}{12}\,\phi''(\xi_i).$$

Аналогично предыдущему получим расчетные формулы:

$$y_{i+1} = y_i + \frac{h}{2}\,\left(f(x_i, y_i) + f(x_{i+1}, y_{i+1})\right), \quad i = 0 : N-1, \quad y_0 = u_a.$$

Это — неявный метод. По поводу его реализации можно сказать то же, что и по поводу реализации неявного метода Эйлера.

Комбинируя этот метод с явной формулой Эйлера с шагом h, получим метод типа предиктор-корректор (пример метода Рунге — Кутты второго порядка точности):

$$y_{i+1} = y_i + \frac{h}{2}\,\left(f(x_i, y_i) + f(x_{i+1}, \overline{y}_{i+1})\right),\ i = 0 : N-1,\ \ y_0 = u_a, \quad (13)$$

где

$$\overline{y}_{i+1} = y_i + hf(x_i, y_i). \quad (14)$$

Шаг (14) называется предиктором, шаг (13) — корректором: сначала по формуле (14) мы выполняем, как бы предсказание (to predict — предсказывать), а затем по формуле (13) уточнение (to correct — исправлять, уточнять) значения y_{i+1}.

4.5. Метод Рунге-Кутта четвертого порядка точности. В основе этого метода лежит квадратурная формула Симпсона

$$\int_{x_i}^{x_{i+1}} \phi(x)\,dx = \frac{h}{6}\,(\phi(x_i) + 4\phi(x_i + h/2) + \phi(x_{i+1}) + \mu_{i+1},$$

$$\mu_{i+1} = -\frac{h^5}{2880}\,\phi^{(4)}(\xi_i).$$

Используются также предсказания по явной формуле Эйлера. Окончательные формулы имеют вид:

$$y_{i+1} = y_i + \frac{h}{6}\,(k_1 + 2k_2 + 2k_3 + k_4)\,, \quad i = 0 : N-1,\ y_0 = u_a,$$

где

$$k_1 = f(x_i, y_i), \qquad k_2 = f\Big(x_i + \frac{h}{2}\,, y_i + h\,\frac{k_1}{2}\Big),$$

$$k_3 = f\Big(x_i + \frac{h}{2}\,, y_i + h\,\frac{k_2}{2}\Big), \qquad k_4 = f\,(x_i + h, y_i + h\,k_3)\,.$$

Здесь каждый шаг требует вычисления четырех значений функции f. Это обеспечивает более высокую точность.

§ 1. Семейство методов Рунге – Кутты.

Явные методы Рунге — Кутты, требующие q вычислений правой части f на одном шаге интегрирования (q-стадийные RK-методы), имеют вид

$$y_{i+1} = y_i + h\big(b_1k_1 + b_2k_2 + \ldots + b_qk_q\big),$$

где

$$\begin{aligned} k_1 &= f(x_i, y_i),\\ k_2 &= f(x_i + c_2h, y_i + a_{21}hk_1),\\ &\ldots \quad \ldots \quad \ldots \quad \ldots \quad \ldots\\ k_q &= f(x_i + c_qh, y_i + a_{q1}hk_1 + a_{q2}hk_2 + \ldots + a_{q,q-1}hk_{q-1}). \end{aligned} \tag{15}$$

При фиксированном q конкретный метод определяется выбором коэффициентов b_j, c_j, a_{ij}. Всегда выполняется равенство

$$b_1 + b_2 + \ldots + b_q = 1, \tag{16}$$

так что при $q = 1$ приходим к методу Эйлера. Коэффициенты этого метода принято располагать в виде следующей треугольной таблицы:

$$\begin{array}{c|ccccc} 0 & & & & & \\ c_2 & a_{21} & & & & \\ c_3 & a_{31} & a_{32} & & & \\ \vdots & \vdots & \vdots & \ddots & & \\ c_q & a_{q1} & a_{q2} & \ldots & a_{q,q-1} & \\ \hline & b_1 & b_2 & \ldots & b_{q-1} & b_q \end{array}$$

Величины k_j, определенные в (15), зависят, в частности, от h и y_i; укажем эту зависимость в виде $k_j = k_j(h, y_i)$. Так, если $u_i = u(x_i)$ есть значение точного решения в узле сетки x_i, то $k_1(h, u_i) = f(x_i, u_i)$, $k_2(h, u_i) = f(x_i + c_2 h, u_i + a_{21} h k_1(x_i, u_i))$, и т.д. Величину

$$\psi_{i+1} = \psi_{i+1}(h) = \frac{u(x_i + h) - u(x_i)}{h} - \sum_{j=1}^{q} b_j \, k_j(h, u_i) \tag{17}$$

называют *погрешностью аппроксимации* в точке сетки x_{i+1}, и рассматривают ее как функцию шага сетки h.

Разложим $\psi_{i+1}(h)$ в ряд Тейлора с остаточным членом в форме Пеано,

$$\psi_{i+1}(h) = \psi_{i+1}(0) + \psi'_{i+1}(0)h + \ldots + \psi^{(m)}_{i+1}(0) \, \frac{h^m}{m!} + o(h^m),$$

и подберем коэффициенты b_j, c_j, a_{ij} так, чтобы при возможно большем значении m выполнялись равенства

$$\psi_{i+1}(0) = \psi'_{i+1}(0) = \ldots = \psi^{(m-1)}_{i+1}(0) = 0. \tag{18}$$

Тогда

$$\psi_{i+1} = \psi^{(m)}_{i+1}(0) \, \frac{h^m}{m!} + o(h^m) = O(h^m), \tag{19}$$

и говорят, что *порядок погрешности аппроксимации* метода равен m, а первое слагаемое в правой части (19) называют главным членом погрешности аппроксимации.

Нетрудно видеть, что условие первого порядка аппроксимации (первое равенство $\psi_{i+1}(0) = 0$ в (18)) совпадает с условием (16), поскольку, в силу уравнения, $k_j(0, u_i) = f(x_i, u_i) = u'(x_i)$,

$$\psi_{i+1}(0) = u'(x_i) - \sum_{j=1}^{q} b_j\, k_j(0, u_i) = \Big(1 - \sum_{j=1}^{q} b_j\Big) u'(x_i).$$

Важность повышения порядка погрешности аппроксимации следует из оценки (см. далее)

$$\|u - y\|_\infty \leqslant C_0 \|\psi\|_\infty, \quad \|y\|_\infty = \max_{1 \leqslant i \leqslant N} |y(x_i)|,$$

с постоянной C_0, не зависящей от h. Из этой оценки следует, что если $\|\psi\|_\infty \leqslant C(u)\, h^m$, то

$$\|u - y\|_\infty \leqslant C_0\, C(u) h^m.$$

В этом случае говорят, что *порядок точности метода* равен m. Как видим, порядок точности совпадает с порядком погрешности аппроксимации, чем и обусловлен способ выбора коэффициентов метода.

§ 2. Двухстадийные методы Рунге — Кутты

Формулы метода при $q = 2$ имеют вид

$$y_{i+1} = y_i + h\big(b_1 k_1 + b_2 k_2\big),$$

где

$$\begin{aligned} k_1 &= f(x_i, y_i), \\ k_2 &= f(x_i + c_2 h, y_i + a_{21} h k_1). \end{aligned}$$

Попытаемся удовлетворить соотношениям (18) при возможно большем значении m. В данном случае

$$\begin{aligned} \psi_{i+1}(h) &= \frac{u(x_i + h) - u(x_i)}{h} - \big(b_1 k_1(h, u_i) + b_2 k_2(h, u_i)\big) = \\ &= \frac{u(x_i + h) - u(x_i)}{h} - \Big(b_1 f(x_i, u_i) + b_2 f\big(x_i + c_2 h, u_i + a_{21} h f(x_i, u_i)\big)\Big) = \\ &= S_1(h) - S_2(h). \end{aligned}$$

Как мы видели выше, равенство $\psi_{i+1}(0) = 0$ выполняется, если

$$b_1 + b_2 = 1.$$

Вычислим $\psi'_{i+1}(0)$. Для этого разложим $u(x_i + h)$ в ряд Тейлора. Получим

$$S_1(h) = \frac{u(x_i + h) - u(x_i)}{h} = u'(x_i) + \frac{h}{2!}\,u''(x_i) + \frac{h^2}{3!}\,u'''(x_i) + \dots \quad (20)$$

Согласно уравнению $u'(x) = f(x, u(x))$. Поэтому

$$u''(x) = f'_x(x, u(x)) + f'_u(x, u(x))u'(x) = \\ = f'_x(x, u(x)) + f'_u(x, u(x))f(x, u(x)),$$

Эти формулы позволяют вычислить $S'_1(0)$:

$$S'_1(0) = 0.5\,u''(x_i) = 0.5\big(f'_x + f'_u f\big)\big|_{x=x_i, u=u_i}.$$

Учитывая, что

$$S'_2(0) = \big(c_2 b_2 f'_x + a_{21} b_2 f'_u\, f\big)\big|_{x=x_i, u=u_i},$$

а также равенство $\psi'_{i+1}(0) = S'_1(0) - S'_2(0)$, из уравнения

$$\psi'_{i+1}(0) = \big((0.5 - c_2 b_2) f'_x + (0.5 - a_{21} b_2) f'_u\, f\big)\big|_{x=x_i, u=u_i} = 0$$

получим еще два уравнения для определения коэффициентов:

$$0.5 - c_2 b_2 = 0, \quad 0.5 - a_{21} b_2 = 0.$$

Следующему условию $\psi''_{i+1}(0) = 0$ выбором коэффициентов удовлетворить нельзя. Это легко увидеть на примере уравнения с $f(x, u) = u$. Тогда $S_2(h)$ линейно по h и его вторая производная по h равна нулю. Из формулы (20) следует, что вторая производная по h от $S_1(h)$ при $h = 0$ равна $1/3u'''(x_i)$. Поэтому

$$\psi''_{i+1}(0) = 1/3\,u'''(x_i).$$

Следовательно, $m = 2$ и четыре коэффициента метода необходимо определить из трех алгебраических уравнений:

$$b_1 + b_2 = 1,$$
$$c_2 b_2 = 0.5,$$
$$a_{21} b_2 = 0.5.$$

Принимая, например, c_2 за свободный параметр $\alpha \in (0, 1]$, найдем:

$$a_{21} = \alpha, \quad b_2 = \frac{1}{2\alpha}, \quad b_1 = 1 - \frac{1}{2\alpha}. \tag{21}$$

Таким образом, мы получили однопараметрическое семейство формул Рунге – Кутты второго порядка аппроксимации. В частности, выбирая $\alpha = 1$ и $\alpha = 1/2$ в (21), придем к двум методам, которые были рассмотрены ранее (центральных прямоугольников и трапеций).

§ 3. Оценка точности методов Рунге — Кутты

Лемма 1. *Пусть для любого $x \in [a, b]$ функция $p \to f(x, p)$ удовлетворяет условию Липшица, т.е. $\exists\, \lambda = \text{const} > 0$ такая, что*

$$|f(x, y) - f(x, z)| \leqslant \lambda\, |y - z| \quad \forall\, y, z \in \mathbb{R}. \tag{22}$$

Тогда функция $k_j(h, y)$ также удовлетворяет условию Липшица:

$$|k_j(h, y) - k_j(h, z)| \leqslant \lambda_j\, |y - z| \quad \forall\, y, z \in \mathbb{R}, \tag{23}$$

где постоянная $\lambda_j > 0$ не зависит от h, $j = 1 : q$.

Доказательство. По индукции. При $j = 1$ оценка (23) выполнена с $\lambda_1 = \lambda$ в силу (22), т.к. $k_1(h, y) = f(x_i, y)$. Пусть (23) выполнено при $j = 1 : s - 1$. Тогда

$$\begin{aligned} |k_s(h, y) - k_s(h, z)| = \Big| f\big(x_i + c_s h, y + \sum_{\ell=1}^{s-1} h\, a_{s\ell}\, k_\ell(h, y)\big) - \\ - f\big(x_i + c_s h, z + \sum_{\ell=1}^{s-1} h\, a_{s\ell}\, k_\ell(h, z)\big) \Big| \leqslant \\ \leqslant \lambda \Big| y - z + \sum_{\ell=1}^{s-1} h\, a_{s\ell}\, \big(k_\ell(h, y) - k_\ell(h, z)\big) \Big| \leqslant \\ \leqslant \lambda \Big(1 + \sum_{\ell=1}^{s-1} (b - a)\, |a_{s\ell}|\, \lambda_\ell \Big)\, |y - z| = \lambda_s\, |y - z|. \quad \square \end{aligned}$$

Следствие 1. *Пусть $F(y) = b_1 k_1(h, y) + \ldots + b_q k_q(h, y)$. Тогда*

$$|F(y) - F(z)| \leqslant \Lambda\, |y - z| \quad \forall\, y, z \in \mathbb{R}, \tag{24}$$

где постоянная $\Lambda > 0$ *не зависит от* h.

Доказательство. Имеем

$$|F(y) - F(z)| = \Big| \sum_{j=1}^{q} b_j \big(k_j(h,y) - k_j(h,y)\big) \Big| \leqslant$$

$$\leqslant \sum_{j=1}^{q} \big|b_j\big| \, \big|k_j(h,y) - k_j(h,y)\big| = \Big(\sum_{j=1}^{q} \big|b_j|\lambda_j \Big) \, |y - z| = \Lambda \, |y - z|. \quad \square$$

Теорема 1. *Пусть* f *удовлетворяет условию* (22). *Тогда*

$$\|u - y\|_\infty \leqslant C \, \|\psi\|_\infty \,,$$

где постоянная $C > 0$ *не зависит от* h.

Доказательство. Пусть $z_i = u_i - y_i$. По определению для $i = 0 : N-1$

$$u_{i+1} = u_i + h \, F(u_i) + h \, \psi_{i+1}, \quad y_{i+1} = y_i + h \, F(y_i).$$

Вычитая из первого равенства почленно второе, получим

$$z_{i+1} = z_i + h \, (F(x_i, u_i) - F(x_i, y_i)) + h\psi_{i+1}. \tag{25}$$

В силу (24) отсюда следует оценка ($1 + x \leqslant e^x \ \forall \, x \geqslant 0$)

$$|z_{i+1}| \leqslant |z_i| + \Lambda \, h \, |z_i| + h|\psi_{i+1}| =$$
$$= (1 + \Lambda \, h)|z_i| + h|\psi_{i+1}| \leqslant e^{\Lambda \, h}|z_i| + h|\psi_{i+1}|.$$

Эта оценка справедлива для любых $i = 0 : N - 1$. Поэтому также

$$|z_i| \leqslant e^{\Lambda \, h}|z_{i-1}| + h|\psi_i|,$$

следовательно,

$$|z_{i+1}| \leqslant e^{2\Lambda \, h}|z_{i-1}| + he^{\Lambda \, h}|\psi_i| + h \, |\psi_{i+1}|.$$

Продолжая аналогичным образом, получим ($z_0 = 0$, $(i+1)h \leqslant b - a$):

$$|z_{i+1}| \leqslant e^{(i+1)\Lambda \, h}|z_0| + h \sum_{k=0}^{i} e^{(i-k)\Lambda \, h}|\psi_{k+1}| \leqslant he^{i \, h\Lambda} \sum_{k=0}^{i} |\psi_{k+1}| \leqslant$$
$$\leqslant (i+1)he^{(i+1)h\Lambda}\|\psi\|_\infty \leqslant (b-a)e^{(b-a)\Lambda} \, \|\psi\|_\infty = C \, \|\psi\|_\infty. \quad \square$$

Следствие 2. *Если метод Рунге – Кутты с q стадиями имеет порядок погрешности аппроксимации, равный m, то*

$$\|u - y\|_\infty \leqslant C\,h^m\,,$$

где постоянная $C > 0$ не зависит от h.

1. Зависимость m от q. Известно, что всегда $m \leqslant q$ и не существует m-стадийных методов с порядком точности m при $m \geqslant 5$. В следующей таблице указано минимальное число стадий ($q_{\min}$), которое необходимо для соответствующего порядка точности метода.

m	1	2	3	4	5	6	7	8
$q_{\min}$	1	2	3	4	6	7	9	11

Время работы T_q программы, реализующей методы с q-стадиями, определяется величиной qT_fN, где T_f время, необходимое для вычисления правой части f уравнения ОДУ при заданных аргументах. При этом максимальная погрешность решения оценивается сверху величиной $C\,h^m \approx C/N^m$. Увеличивая число узлов, скажем, в 10 раз, мы получаем в 10^m раз более точное решение. Отсюда ясно, в чем выгода от использования методов высокого порядка точности.

С другой стороны, если в вычислениях не нужна очень большая точность, то методы со средним показателем m (например, $m = 4$), могут оказаться предпочтительными, поскольку они требуют меньшего времени работы программы и имеют достаточную точность.

2. О вложенных методах Рунге – Кутты. Рассмотрим q-стадийный явный метод Рунге — Кутты, определяемый следующей таблицей:

0					
c_2	a_{21}				
c_3	a_{31}	a_{32}			
$\vdots$	$\vdots$	$\vdots$	$\ddots$		
c_q	a_{q1}	a_{q2}	$\dots$	$a_{q,q-1}$	
	b_1	b_2	$\dots$	b_{q-1}	b_q
	$\hat{b}_1$	$\hat{b}_2$	$\dots$	$\hat{b}_{q-1}$	$\hat{b}_q$

Эта таблица имеет два набора коэффициентов $b : \{b_i\}$ и $\{\hat{b}_i\}$. Предположим, что приближенное решение y_i в точке x_i найдено с требуемой точностью и решение в точке $x_{i+1} = x_i + h$ находится по формуле

$$y_{i+1} = y_i + h\big(b_1k_1 + b_2k_2 + \ldots + b_qk_q\big), \tag{26}$$

имеющей порядок точности равный m. Здесь k_i вычисляются согласно формулам (15). Пусть также коэффициенты $\{\hat{b}_i\}$ таковы, что

$$\hat{y}_{i+1} = y_i + h\big(\hat{b}_1k_1 + \hat{b}_2k_2 + \ldots + \hat{b}_qk_q\big) \tag{27}$$

другое решение, но порядка точности $m + 1$ (k_i в (27) те же, что и в (26)). При этих предположениях за q вычислений правой части одновременно получаются два приближенных решения, точность которых отличается на порядок по h.

В силу малости h, величину $err_{i+1} = |y_{i+1} - \hat{y}_{i+1}|/|\hat{y}_{i+1}|$ можно принять за меру локальной относительной погрешности решения y_{i+1} в точке x_{i+1}; $\hat{y}_{i+1}$ используется при этом лишь для определения этой погрешности.

Вложенные методы обычно называют по фамилии автора с указанием порядка основного и вспомогательного метода; например, метод Мерсона 4(5). На практике широко используются различные вложенные методы, среди которых выделяются методы Дормана–Принса, известные при различных m. Например, формулы Дормана–Принса 4(5) определяются следующей таблицей:

0							
$\frac{1}{5}$	$\frac{1}{5}$						
$\frac{3}{10}$	$\frac{3}{40}$	$\frac{9}{40}$					
$\frac{4}{5}$	$\frac{44}{45}$	$-\frac{56}{15}$	$\frac{32}{9}$				
$\frac{8}{9}$	$\frac{19372}{6561}$	$-\frac{25360}{2187}$	$\frac{64448}{6561}$	$-\frac{212}{729}$			
1	$\frac{9017}{3168}$	$-\frac{355}{33}$	$\frac{46732}{5247}$	$\frac{49}{176}$	$-\frac{5103}{18656}$		
1	$\frac{35}{384}$	0	$\frac{500}{1113}$	$\frac{125}{192}$	$-\frac{2187}{6784}$	$\frac{11}{84}$	
y	$\frac{35}{384}$	0	$\frac{500}{1113}$	$\frac{125}{192}$	$-\frac{2187}{6784}$	$\frac{11}{84}$	0
$\hat{y}$	$\frac{5179}{57600}$	0	$\frac{7571}{16695}$	$\frac{393}{640}$	$-\frac{92097}{339200}$	$\frac{187}{2100}$	$\frac{1}{40}$

Этот метод имеет семь стадий, однако практически — шесть (как отмечалось ранее, метод пятого порядка не может иметь меньше,

чем шесть стадий). Действительно, поскольку $a_{7,i} = b_i$ для всех i, то нетрудно видеть, что коэффициент k_7 на предыдущем шаге совпадает с коэффициентом k_1 на следующем шаге. Этот факт используется при программировании и позволяет экономить одно вычисление f.

3. Об автоматическом выборе шага. Ранее мы рассмотрели RK-методы с постоянным шагом интегрирования h. Однако ясно, что h можно выбирать на каждом шаге интегрирования, например, автоматически, используя то или иной алгоритм.

Рассмотрим идею процедуры автоматического выбора шага при использовании вложенных методов типа $m(m+1)$. Пусть решение в точке x_i найдено с некоторым h. Выполняется следующий цикл.

1) С использованием вложенных формул вычисляется y_{i+1} и величина погрешности $err = |y_{i+1} - \hat{y}_{i+1}|/|\hat{y}_{i+1}|$.

2) Если $err \leqslant tol$, т. е. погрешность меньше или равна заданной допустимой погрешности, то решение в точке x_{i+1} считается найденным, а шаг интегрирования — выполненным успешно. Далее вычисляется новый шаг интегрирования $h = \min(\gamma\, h, hmax)$, где $\gamma \geqslant 1$, и h принимается в качестве текущего для следующего шага и интегрирование продолжается. Здесь $hmax$ —максимальный шаг (задается пользователем);

3) Если $err > tol$, т. е. погрешность оказывается больше желаемой, то полученное решение в точке x_{i+1} отбрасывается как неточное, а шаг интегрирования считается неудачным. Вычисляется новый шаг равный $\gamma\, h$, где теперь $\gamma < 1$, и вычисления повторяются с шага 1).

Для выбора γ, как правило, используют формулу

$$\gamma = \min(facmax, \max(facmin, fac\,(tol/err)^{1/(m+1)}).$$

Здесь $facmax$ ($facmin$) — максимальный коэффициент увеличения (уменьшения) шага ($facmax = 1.5, \ldots, 5$). Эти коэффициенты страхуют от резкого увеличения или уменьшения шага. Для шагов, выполняемых непосредственно после отбрасывания решения, рекомендуется выбирать $facmax = 1$; fac — страховочный коэффициент ($fac = 0.8, \ldots, 0.9$).

Большинство современных программ для ЭВМ для решения задачи Коши автоматически выбирают шаг интегрирования. Это обеспечивает изменение шага h в зависимости от скорости изменения ис-

комого решения. Там, где решение меняется плавно, программа автоматически переходит на более крупный шаг и, наоборот, при более быстром изменении решения шаг сетки уменьшается. Это позволяет существенно ускорять решение задачи, не снижая точности.

Задания для самостоятельной работы

ВОПРОСЫ ДЛЯ САМОКОНТРОЛЯ

1. Укажите способ, позволяющий свести ОДУ 4-го порядка к системе 4-х уравнений первого порядка.

2. Дайте формулировку задачи Коши для системы ОДУ. В чем ее отличие от краевой задачи?

3. В чем заключается метод разложения решения в ряд Тейлора? Как получаются расчетные формулы? В чем состоит основной недостаток этого метода?

4. Дайте общую характеристику одношаговых и многошаговых сеточных методов.

5. Приведите формулы явного метода Эйлера. Укажите оценку его погрешности аппроксимации.

6. Приведите формулы неявного метода Эйлера. Укажите оценку его погрешности аппроксимации.

7. Укажите возможные способы определения решения (реализации) неявного метода Эйлера.

8. Приведите формулы метода предиктор-корректор.

9. Дайте определение семейства q-стадийных методов Рунге – Кутты.

10. Дайте определение а) погрешности аппроксимации q-стадийного метода Рунге – Кутты. б) какое число называется порядком погрешности аппроксимации?

11. Укажите способ определения коэффициентов (параметров) метода Рунге – Кутты.

12. Дайте определение порядка точности сеточного метода.

13. Сформулируйте теорему о точности метода Рунге – Кутты.

14. Существуют ли q-стадийные методы Рунге – Кутты точности m при: a) $q = 1$, $m = 1$? b) $q = 21$, $m = 2$? c) $q = 3$, $m = 3$? d) $q = 4$, $m = 4$? e) $q = 5$, $m = 5$?

15. Какие методы называются вложенными методами Рунге – Кутты?

16. Что понимается под автоматическим выбором шага интегрирования?

17. Как на основе вложенных методов Рунге – Кутты можно организовать автоматический выбор шага интегрирования?

18. В чем смысл методов с автоматическим выбором шага интегрирования?

ЗАДАЧИ И УПРАЖНЕНИЯ

1. Докажите, что не существует q-стадийных методов Рунге – Кутты, порядок погрешности аппроксимации которых равен $q + 1$.

Указание. Рассмотрите подробнее способ построения 2-стадийных методов.

2. Постройте вложенный метод Рунге – Кутты точности 1(2).

Указание. Обратите внимание на метод трапеций.

3. В условиях теоремы 1 о точности метода Рунге – Кутты докажите, что для неявного метода Эйлера справедлива оценка точности $\|u-y\|_\infty \leqslant T\,\|\psi\|_\infty$, если дополнительно выполнена оценка монотонности f по второй переменной:

$$(f(x,u)-f(x,y))(u-y)\leqslant 0 \quad \forall\, x\in[a,b],\ u,y\in\mathbb{R}. \tag{28}$$

Сравните эту оценку с оценкой точности явного метода Эйлера. Чем можно объяснить существенное различие оценок?

Указание. Следуйте доказательству теоремы 1 и пытайтесь учесть дополнительное условие (28).

§ 4. Многошаговые методы

Рассмотренные нами методы Рунге — Кутты имеют вид

$$\frac{y_{i+1}-y_i}{h}=F(y_i),\quad i=0:N-1,$$

и относятся к классу одношаговых методов. В подобных методах после того как найдено очередное значение y_{i+1} в точке x_{i+1} значение y_i отбрасывают и уже не используют в последующих вычислениях.

Естественно попытаться извлечь определенную пользу из информации о значениях решения не в одной, а в нескольких предыдущих точках. Такие методы принято называть многошаговыми методами.

Рассмотрим одно семейство подобных методов.[1)]

1. Методы Адамса. Методы Адамса имеют вид

$$\frac{y_{i+1}-y_i}{h}=\beta_0\,f(x_{i+1},y_{i+1})+\beta_1\,f(x_i,y_i)+\ldots+\beta_k f(x_{i+1-k},y_{i+1-k}), \tag{29}$$

или короче

$$\frac{y_{i+1}-y_i}{h}=\sum_{j=0}^{k}\beta_j\,f_{i+1-j},\quad i=k-1,k,\ldots,N-1, \tag{30}$$

Здесь β_j — числовые коэффициенты, $f_{i+1-j}=f(x_{i+1-j},y_{i+1-j})$.

Уравнение (30) позволяет найти новое значение y_{i+1}, используя найденные ранее значения $y_i,\ y_{i-1},\ \ldots,\ y_{i+1-k}$. Поэтому для начала вычислений требуется задание начальных значений $y_0,\ y_1,\ \ldots,\ y_{k-1}$.

[1)] В честь предложившего их в 1855 году английского математика-астронома Д.К. Адамса.

Они должны быть вычислены каким-либо другим методом с требуемой точностью.

В случае $\beta_0 = 0$ метод Адамса называется явным, так как значение y_{i+1} в этом случае выражается через найденные ранее значения по явной формуле (см. (30)). Если $\beta_0 \neq 0$, то метод Адамса называется неявным.

Существуют различные способы, позволяющие определить коэффициенты β_j в методах типа Адамса. Опишем один такой способ.

2. Определение коэффициентов $\boldsymbol{\beta}$. Как и в методе Рунге — Кутты определим погрешность аппроксимации

$$\psi_{i+1} = \frac{u_{i+1} - u_i}{h} - \sum_{j=0}^{k} \beta_j f(x_{i+1-j}, u_{i+1-j})$$

и подберем β_j так, чтобы обеспечить как можно более высокий порядок ее малости относительно шага h. Т.к. $f(x, u(x)) = u'(x)$, то

$$\psi_{i+1} = \frac{u_{i+1} - u_i}{h} - \sum_{j=0}^{k} \beta_j \, u'(x_{i+1-j}).$$

Воспользуемся формулой Тейлора

$$u_i = u(x_{i+1} - h) = u_{i+1} - hu'_{i+1} + \frac{h^2}{2!} u''_{i+1} - \frac{h^3}{3!} u'''_{i+1} + \cdots,$$

откуда

$$\frac{u_{i+1} - u_i}{h} = u'_{i+1} - \frac{h}{2!} u''_{i+1} + \frac{h^2}{3!} u'''_{i+1} + \cdots$$

Далее,

$$u'(x_{i+1-j}) = u'(x_{i+1} - jh) = u'(x_{i+1}) - jh\, u''(x_{i+1}) + \frac{(jh)^2}{2!} u'''(x_{i+1}) + \cdots$$

Нетрудно подсчитать теперь, что

$$\psi_{i+1} = u'_{i+1}\Big(1 - \sum_{j=0}^{k} \beta_j\Big) - \frac{h}{2!} u''_{i+1}\Big(1 - 2\sum_{j=0}^{k} j\beta_j\Big) + \\ + \frac{h^2}{3!} u'''_{i+1}\Big(1 - 3\sum_{j=0}^{k} j^2\beta_j\Big) - \cdots$$

Приравнивая нулю коэффициенты при степенях h, получим СЛАУ относительно $k+1$ неизвестной $\beta_0, \dots \beta_k$:

$$\sum_{j=0}^{k} \beta_j = 1, \quad \sum_{j=0}^{k} j\beta_j = 1/2, \quad \dots, \quad \sum_{j=0}^{k} j^{q-1}\beta_j = 1/q. \qquad (31)$$

Понятно, что если эти уравнения будут удовлетворены, то метод будет иметь погрешность аппроксимации порядка q, т.е. $\psi_{i+1} = O(h^q)$.

Если мы хотим получить неявный метод, нужно положить $q = k+1$. В этом случае мы получим СЛАУ, имеющую единственное решение, поскольку определитель матрицы этой системы является определителем Вандермонда (убедитесь в этом!). Для построения явного метода, нужно положить $\beta_0 = 0$ и рассмотреть СЛАУ из k уравнений, полагая $q = k$.

Обратим внимание, что при указанных способах выбора q, СЛАУ определяется только параметром k. Решая СЛАУ при заданном $k = 1, 2, \dots$, мы можем раз и навсегда определить β_0, β_1, $\dots$, β_k как для явных, так и неявных методов Адамса.

Из теоремы 2 следует, что порядок точности метода Адамса (как явной, так и неявной) равен порядку погрешности аппроксимации.

Теорема 2. (без доказательства) *Пусть начальные условия метода Адамса определены с точностью порядка погрешности аппроксимации, то есть* $y_i = u_i + O(h^q)$ *для* $i = 0 : k-1$, $\psi_i = O(h^q)$, $i = k-1 : N-1$, *функция* $f(x,p)$ *удовлетворяет условию Липшица по переменной* p *при каждом* $x \in [a,b]$. *Тогда* $|u(x_i) - y_i| = O(h^q)$, $i = 0 : N$. *Иными словами,* q *есть порядок точности метода.*

§ 5. Явные методы Адамса

При $\beta_0 = 0$ из (30) получаем

$$y_{i+1} = y_i + h\sum_{j=1}^{k} \beta_j f(x_{i+1-j}, y_{i+1-j}), \quad i = k-1, k, \dots N-1, \qquad (32)$$

При $k = 1$ метод совпадает с явным методом Эйлера (см. (31)).

Важно отметить, что при переходе от y_i к y_{i+1} требуется вычислить лишь одно новое значение функции f, а именно $f(x_i, y_i)$, все остальные значения функции f, входящие в правую часть (32), могут

быть извлечены из памяти ЭВМ, если на предыдущих шагах позаботиться об их сохранении. Итак, по сравнению с q-стадийным методом Рунге — Кутты вычислений на каждом шагу требуется в q раз меньше. Это несомненное достоинство методов типа Адамса.

Для определения коэффициентов метода, можно решить систему уравнений (31) при $q = k$ и $\beta_0 = 0$. Имеется также другой явный способ их определения (следуя Д. Адамсу). Укажем его.

Интегрируя уравнение $u' = f(x, u)$ по интервалу $[x_i, x_{i+1}]$ получим

$$\frac{u(x_{i+1}) - u(x_i)}{h} = \frac{1}{h}\int_{x_i}^{x_{i+1}} f(x, u(x))\,dx. \tag{33}$$

Подынтегральную функцию $f(x, u(x))$ аппроксимируем на отрезке $[x_{i+1-k}, x_i]$ интерполяционным полиномом $L_{k-1}(x)$ степени $k-1$ по таблице ее значений в тех узлах, где значения решения уже найдены:

$$\begin{array}{cccc} x_{i+1-k} & \dots & x_{i-1} & x_i \\ f(x_{i+1-k}, u_{i+1-k}) & \dots & f(x_{i-1}, u_{i-1}) & f(x_i, u_i) \end{array}$$

Согласно формуле Лагранжа имеем

$$f(x, u(x)) = L_{k-1}(x) + R_k = \sum_{j=1}^{k} f(x_{i+1-j}, u_{i+1-j})\varphi_j(x) + R_k, \tag{34}$$

$$R_k = \frac{1}{k!}\frac{d^k}{dx^k} f(x, u(x))\Big|_{x=\xi_i(x)} \omega_k(x) = \frac{1}{k!}\, u^{(k+1)}(\xi_i(x)))\,\omega_k(x)\,,$$

где

$$\omega_k(x) = (x - x_{i+1-k})\dots(x - x_{i-1})(x - x_i),$$

$$\varphi_j(x) = \prod_{s=1, s\neq j}^{k} \frac{x - x_{i+1-s}}{x_{i+1-j} - x_{i+1-s}}\,.$$

Используя (34) в (33), придем к следующим соотношениям:

$$\frac{u(x_{i+1}) - u(x_i)}{h} = \sum_{j=1}^{k} \beta_j f(x_{i+1-j}, u_{i+1-j}) + \psi_{i+1}, \tag{35}$$

$$\beta_j = \frac{1}{h}\int_{x_i}^{x_{i+1}} \varphi_j(x)\,dx, \quad \psi_{i+1} = \frac{1}{h\,k!}\int_{x_i}^{x_{i+1}} u^{(k+1)}(\xi_i(x))\,\omega_k(x)\,dx.$$

Для $x \in [x_i, x_{i+1}]$ имеем $|\omega_k(x)| \leqslant k!\, h^k$. Поэтому

$$|\psi_{i+1}| \leqslant M_{k+1} h^k, \quad M_{k+1} = \max_{x\in[a,b]} |u^{(k+1)}(x)|.$$

Отбрасывая в (35) малое слагаемое ψ_{i+1} и заменяя $u(x)$ на приближение $y(x)$, получим явную схему Адамса (32), для коэффициентов которого получено явное выражение.

Замечание 1. Так как интерполяционный полином используется для приближения функции вне отрезка, на котором он определен, то в действительности метод основан на экстраполяции. Поэтому явный метод Адамса называют также экстраполяционным методом Адамса.

Замечание 2. Так как $\sum_{j=1}^{k} \varphi_j(x) \equiv 1$, то из (5) получаем $\sum_{j=1}^{k} \beta_j = 1$, т.е. (31).

Замечание 3. Для вычисления интеграла, определяющего β_j, используем замену переменных $x = x_{i+1} - th$. Тогда $x_{i+1-j} = x_{i+1} - jh$, $j = 0 : k$, $[x_i, x_{i+1}] \to [1, 0]$, $dx = -h\, dt$,

$$\beta_j = \frac{1}{h} \int_{x_i}^{x_{i+1}} \prod_{s=1, s\neq j}^{k} \frac{x - x_{i+1-s}}{x_{i+1-j} - x_{i+1-s}}\, dx = \int_0^1 \prod_{s=1, s\neq j}^{k} \frac{t-s}{j-s}\, dt.$$

Рассмотрим, для примера, вычисление коэффициентов β при $k = 2$. Используя первый способ, мы должны решить систему

$$\beta_1 + \beta_2 = 1, \quad \beta_1 + 2\beta_2 = 1/2 \quad \Rightarrow \quad \beta_1 = 3/2, \quad \beta_2 = -1/2.$$

Используя (5), также получаем

$$\beta_1 = \int_0^1 \frac{t-2}{1-2}\, dt = 3/2, \quad \beta_2 = \int_0^1 \frac{t-1}{2-1}\, dt = -1/2.$$

Приведем формулы явного метода Адамса при $k = 2, 3, 4$:

$$y_{i+1} = y_i + \frac{1}{2}(3f_i - f_{i-1}), \quad k = 2,$$
$$y_{i+1} = y_i + \frac{1}{12}\left(23f_i - 16f_{i-1} + 5f_{i-2}\right), \quad k = 3,$$
$$y_{i+1} = y_i + \frac{1}{24}\left(55f_i - 59f_{i-1} + 37f_{i-2} - 9f_{i-3}\right), \quad k = 4,$$

где $f_i = f(x_i, y_i)$.

§ 6. Невные методы Адамса

К этому классу относятся методы с $\beta_0 \neq 0$. В этом случае для определения y_{i+1} получаем уравнение

$$y_{i+1} = h\beta_0 f(x_{i+1}, y_{i+1}) + F_i, \quad F_i = h\sum_{j=1}^{k} \beta_j f(x_{i+1-j}, y_{i+1-j}),$$

где F_i вычисляется явно по известным значениям $y(x)$.

Для отыскания y_{i+1}, обычно, используют метод простой итерации

$$y_{i+1}^{(s+1)} = h\beta_0 f(x_{i+1}, y_{i+1}^{(s)}) + F_i, \quad s = 0, 1, \ldots$$

В качестве $y_{i+1}^{(0)}$ можно использовать значение y_{i+1}, полученное некоторым явным методом Адамса (например, того же порядка точности).

Для коэффициентов метода, которые можно определить решая СЛАУ (31), можно получить явные формулы, как это было сделано ранее для явных методов Адамса.

А именно, функцию $f(x, u(x))$ на отрезке $[x_{i+1-k}, x_{i+1}]$ аппроксимируем интерполяционным полиномом $L_k(x)$ степени k по таблице

x_{i+1-k}	$\ldots$	x_i	x_{i+1}
$f(x_{i+1-k}, u_{i+1-k})$	$\ldots$	$f(x_i, u_i)$	$f(x_{i+1}, u_{i+1})$

Далее, рассуждая аналогично явному методу, придем к формулам

$$\beta_j = \frac{1}{h}\int_{x_i}^{x_{i+1}} \prod_{s=0, s\neq j}^{k} \frac{x - x_{i+1-s}}{x_{i+1-j} - x_{i+1-s}}\, dx = \int_0^1 \prod_{s=0, s\neq j}^{k} \frac{t-s}{j-s}\, dt, \qquad (36)$$

$$|\psi_{i+1}| \leqslant \frac{M_{k+1}}{k+1}\, h^{k+1}.$$

Замечание 4. Неявный метод Адамса называют также интерполяционным методом Адамса, поскольку при построении метода интегрирование интерполяционного полинома производится по отрезку, на котором он был определен.

Рассмотрим, примеры неявных методов.

1). При $k = 0$ из (31) получаем $\beta_0 = 1$. В этом случае метод Адамса совпадает с неявным методом Эйлера:

$$y_{i+1} = y_i + h\, f(x_{i+1}, y_{i+1}).$$

2). При $k = 1$, используя первый способ, получаем

$$\beta_0 + \beta_1 = 1, \quad \beta_2 = 1/2 \quad \Rightarrow \quad \beta_1 = 1/2, \quad \beta_2 = 1/2.$$

Используя (36), также получаем

$$\beta_1 = \int_0^1 \frac{t-1}{0-1}\,dt = 1/2, \quad \beta_2 = \int_0^1 \frac{t-0}{1-0}\,dt = 1/2.$$

Поэтому неявный метод Адамса второго порядка точности имеет вид

$$y_{i+1} = y_i + h/2\left(f(x_{i+1}, y_{i+1}) + f(x_i, y_i)\right)$$

и совпадает с методом трапеций.

3). При $k = 2$ получаем систему

$$\begin{aligned} \beta_0+\beta_1 + \beta_2 &= 1, \\ \beta_1 + 2\beta_2 &= 1/2, \\ \beta_1 + 4\beta_2 &= 1/3. \end{aligned}$$

Нетрудно получить: $\beta_2 = -1/12$, $\beta_1 = 2/3$, $\beta_0 = 5/12$. Поэтому неявный метод Адамса третьего порядка точности имеет вид

$$\frac{y_{i+1} - y_i}{h} = \frac{1}{12}\left(5f\left(x_{i+1}, y_{i+1}\right) + 8f\left(x_i, y_i\right) - f\left(x_{i-1}, y_{i-1}\right)\right).$$

§ 7. Устойчивость явных и неявных методов

Возникает естественный вопрос: зачем нужны неявные методы, если есть более простые явные методы? Ответ на этот вопрос связан с понятием устойчивости методов решения задачи Коши.

В приложениях довольно часто возникают уравнения с асимптотически устойчивыми решениями, удовлетворяющими условию:

$$|u(x+\tau)| \leqslant |u(x)| \quad \forall\, \tau > 0.$$

Модельным примером такого уравнения является простое уравнение

$$u'(x) = -\lambda u(x), \quad x > 0,\ \lambda > 0.$$

Его решение

$$u(x) = u(0)e^{-\lambda x}.$$

Желательно, чтобы и приближенное решение было асимптотически устойчивым, то есть были выполнены неравенства

$$|y_{i+1}| \leqslant |y_i| \quad \forall\, i \geqslant 0. \tag{37}$$

Рассмотрим с этой точки зрения два простейших метода.

1. Явный метод Эйлера. Применительно к рассматриваемой задаче он принимает вид:

$$\frac{y_{i+1} - y_i}{h} = -\lambda y_i, \quad y_0 = u(0). \tag{38}$$

Имеем:

$$y_{i+1} = (1 - \lambda h) y_i, \quad i = 0 : N-1 \quad \Rightarrow \quad y_i = (1 - \lambda h)^i\, u(0).$$

Так как $|y_{i+1}| \leqslant |1 - \lambda h|\, |y_i|$, то неравенства (37) выполнены, если

$$h \leqslant \frac{2}{\lambda}\,. \tag{39}$$

Если это условие (условие устойчивости явного метода Эйлера) не выполнено, т.е. $h > 2/\lambda$, то y_i меняют знак и $|y_i| = |y(x_i)|$ возрастают с ростом i, что в корне отличается от поведения решения $u(x)$, которое положительно и убывает с ростом x.

2. Неявный метод Эйлера. Имеем

$$\frac{y_{i+1} - y_i}{h} = -\lambda y_{i+1}, \quad y_0 = u(0), \tag{40}$$

или $y_{i+1} = y_i/(1 + \lambda h)$ для всех $i = 0 : N-1$. Следовательно

$$y_i = \frac{1}{(1 + \lambda h)^i}\, u(0).$$

Ясно, что y_i положительны и монотонно убывают, а неравенства (37) выполнены при любом h.

Метод (40) называют *абсолютно устойчивым*, а метод (38) — *условно устойчивым*. Применение условно устойчивого метода может потребовать слишком малого шага (при $\lambda \gg 1$) для обеспечения устойчивости метода и, следовательно, большой вычислительной работы. Выбор шага абсолютно устойчивого метода подчинен лишь соображениям точности.

В рассмотренном нами примере при больших λ решение резко меняется на начальном участке (производная велика по модулю), затем начинает выполаживаться (при больших x производная мала). Поэтому вполне естественно сначала вести счет с малым шагом, а затем шаг увеличивать. Абсолютно устойчивый метод позволяет это сделать, Если же мы попытаемся увеличивать шаг в явном методе и нарушим условие (39), то величина $1 - \lambda h$ будет меньше -1 и решение начнет "разбалтываться". Задачи, подобные рассмотренной нами, имеющие участки резкого изменения решения, принято называть жесткими. Приведенный пример показывает, что для их приближенного решения целесообразно использовать неявные методы.

Задания для самостоятельной работы

ВОПРОСЫ ДЛЯ САМОКОНТРОЛЯ

1. Дайте определение явных методов Адамса.

2. Дайте определение погрешности аппроксимации явных методов Адамса. Укажите способы определения его коэффициентов.

3. Почему явные методы Адамса называются также экстраполяционными?

4. Приведите формулы явного метода Адамса первого порядка аппроксимации.

5. Сравните явные методы Адамса и Рунге – Кутты одинакового порядка аппроксимации. Какие формулы экономичнее.

6. Дайте определение неявных методов Адамса.

7. Дайте определение погрешности аппроксимации неявных методов Адамса. Укажите способы определения его коэффициентов.

8. Почему неявные методы Адамса называются также интерполяционными?

9. Приведите формулы неявного метода Адамса первого порядка аппроксимации.

10. Укажите возможные способы определения решения (реализации) неявных методов Адамса.

11. Как можно определить методы типа предиктор-корректор на основе явных и неявных методов Адамса?

12. зачем нужны неявные методы, если есть более простые явные методы?

13. Охарактеризуйте устойчивость явных и неявных методов Адамса на примере модельной задачи. Почему дополнительное условие устойчивости явных схем может оказаться ограничительным?

ЗАДАЧИ И УПРАЖНЕНИЯ

1. Получите формулы явного метода Адамса второго порядка аппроксимации.

2. Получите формулы явного метода Адамса третьего порядка аппроксимации.

3. Получите формулы явного метода Адамса четвертого порядка аппроксимации.
4. Получите формулы неявного метода Адамса второго порядка аппроксимации.
5. Получите формулы неявного метода Адамса третьего порядка аппроксимации.
6. Получите формулы неявного метода Адамса четвертого порядка аппроксимации.

ГЛАВА 6

Методы решения одномерных краевых задач

Рассмотрим несколько приближенных методов решения задачи

$$-(pu')' + qu = f, \quad 0 < x < \ell, \tag{1}$$

$$u(0) = u(\ell) = 0. \tag{2}$$

Здесь p, q, f — заданные достаточно гладкие функции, причем

$$0 < \alpha \leqslant p\,(x) \leqslant \beta, \quad q(x) \geqslant 0 \quad \forall\, x \in [0, \ell], \tag{3}$$

где $\alpha, \beta = \text{const}$. В отличие от задачи Коши дополнительные условия заданы в двух граничных точках отрезка интегрирования, поэтому задача называется граничной или чаще — краевой. Условий (3) достаточно для существования ее единственного решения.

Задача (1), (2) может быть, например, интерпретирована как задача о стационарном распределении температуры в стержне, боковая поверхность которой имеет заданную температуру.

Традиционный интерес к этой задаче обусловлен не столько важностью ее приложений, сколько тем, что задача (1), (2) может рассматриваться как хорошая модель для отработки методов численного решения более сложных двумерных и трехмерных эллиптических уравнений.

§ 1. Метод коллокации

Уравнение (1) запишем в виде

$$u''(x) + v(x)u'(x) + g(x)u(x) = F(x), \quad x \in (0, \ell), \tag{4}$$

где $v = p'$, $g = -q/p$, $F = -f/p$.

Приближенное решение задачи будем искать в виде полинома $y(x) \in P_{n-1}$, удовлетворяющего краевым условиям $y(0) = y(\ell) = 0$. С этой целью на отрезке $[0, \ell]$ введем сетку узлов

$$0 = x_1 < x_2 < \ldots < x_n = \ell.$$

Положим $y_i = y(x_i)$, $i = 1:n$. Согласно формуле Лагранжа

$$y(x) = \sum_{j=2}^{n-1} y_j\, \varphi_j(x)\,, \quad \varphi_j(x) = \frac{\omega_n(x)}{(x-x_j)\omega_n'(x_j)}\,, \tag{5}$$

где $\omega_n(x) = \prod_{k=1}^{n}(x-x_k)$. Потребуем, чтобы $y(x)$ удовлетворял дифференциальному уравнению, но не во всех точках $(0,\ell)$ (это невозможно), а только во внутренних точках сетки. Получим уравнения

$$\sum_{j=2}^{n-1} \varphi_j''(x_i)\, y_j + \sum_{j=2}^{n-1} v(x_i)\, \varphi_j'(x_i)\, y_j + g(x_i) y_i = F(x_i), \quad i = 2:n-1, \tag{6}$$

которые в матричном виде запишем как $Ay = F$.

Укажем удобный способ вычисления элементов матрицы A. Определим квадратные матрицы $D^{(k)} = \{\varphi_j^{(k)}(x_i)\}_{i,j=1}^{n}$ размера n (матрицы дифференцирования). Из (5) следует, что

$$d_{ij}^{(1)} = \varphi_j'(x_i) = \frac{\omega_n'(x_i)}{(x_i-x_j)\omega_n'(x_j)} = \frac{\beta_j/\beta_i}{x_i-x_j}\,, \quad i \neq j, \tag{7}$$

где $\beta_i = C/\omega_n'(x_i)$ — барицентрические веса. Так как $\sum_{k=1}^{n} \varphi_k(x) = 1$ для всех x, то дифференцированием найдем диагональные элементы

$$d_{ii}^{(1)} = -\sum_{k=1,\,k\neq i}^{n} d_{ik}^{(1)}. \tag{8}$$

Из формулы $\varphi_j'(x) = \sum_{k=1}^{n} \varphi_j'(x_k)\varphi_k(x)$ следует, что $D^{(2)} = \big(D^{(1)}\big)^2$. Таким образом,

$$A = \Big\{ d_{ij}^{(2)} + v(x_i)\, d_{ij}^{(1)} + g(x_i)\delta_{ij} \Big\}_{i,j=2}^{n-1}.$$

После решения системы $Ay = F$ приближение $y(x)$ в любой точке $x \in (0,\ell)$ можно экономно вычислить по барицентрической формуле

$$y(x) = \Big(\sum_{i=2}^{n-1} \frac{\beta_i\, y_i}{x-x_i} \Big) \Big/ \Big(\sum_{i=1}^{n} \frac{\beta_i}{x-x_i} \Big). \tag{9}$$

Свойства матрицы A, а также точность метода, существенно зависят от выбора узлов коллокации $\{x_i\}_{i=1}^{n}$. Как правило, их выбирают

как корни ортогональных полиномов. В этом случае также имеются явные формулы для барицентрических весов. Например, если

$$x_i = \ell \sin^2\Big(\frac{(i-1)\pi}{2(n-1)}\Big), \quad i = 1:n, \tag{10}$$

экстремумы полинома Чебышева T_{n-1}, сдвинутые на $[0,\ell]$, то C можно выбрать так, что $\beta_i = (-1)^i\gamma_i$, где $\gamma_1 = \gamma_n = 1/2$, $\gamma_i = 1$ при $i = 2:n-1$. Отметим, что с ростом n узлы (10) сгущаются к $x = 0, \ell$.

В случае узлов (10) для максимальной погрешности приближенного решения справедлива оценка

$$E = \max_{x\in[0,\ell]} |u(x) - y(x)| \leqslant C\, n^{-s},$$

где постоянная C не зависит от n, а $s = \min(n, m)$, где m — число непрерывных производных, которыми обладают функции p', q и f на отрезке $[0, l]$. Поэтому при гладких исходных данных этот метод имеет высокую точность уже при небольших значениях n.

§ 2. Методы конечных разностей

Методы конечных разностей (МКР) позволяют найти приближение к решению $u(x)$ задачи (1), (2) на некотором множестве точек ω_h. Для упрощения изложения рассмотрим равномерную сетку

$$\omega_h = \{x_i = (i-1)h,\ i = 1:n,\ h = \ell/(n-1)\}\,.$$

Через $y_i = y(x_i)$ обозначим приближенное решение в точке сетки x_i, через $y = (y_1, y_2, \ldots, y_n)^T$ — вектор узловых значений $y(x)$.

Для определения этого вектора в МКР строится СЛАУ $Ay = F$, которая называется *конечно-разностной схемой* решения задачи (1), (2). Рассмотрим один из методов построения таких схем, который называется *методом баланса*.

1. Аппроксимация уравнения. Нам понадобятся приближенные формулы вычисления производных и интегралов:

$$u'\Big(x + \frac{h}{2}\Big) = \frac{u(x+h) - u(x)}{h} + \frac{h^2}{24}\,u'''(\xi(x)), \tag{11}$$

$$\int_a^b \phi(x)\,dx = (b-a)\,\phi\Big(\frac{a+b}{2}\Big) + \frac{(b-a)^3}{24}\,\phi''(\xi). \tag{12}$$

где $\xi(x) \in [x, x+h]$, $\xi \in [a, b]$. Для сокращения записи положим

$$x_{i\pm1/2} = x_i \pm \frac{h}{2}, \quad f_k = f(x_k).$$

В каждой внутренней точке x_i, $i = 2 : n-1$, получим аппроксимацию уравнения (1). Для этого проинтегрируем его по отрезку $[x_{i-1/2}, x_{i+1/2}]$, а результат поделим на h. Получим

$$-\frac{1}{h}\left(p_{i+1/2}\,u'_{i+1/2} - p_{i-1/2}\,u'_{i-1/2}\right) + \frac{1}{h}\int_{x_{i-1/2}}^{x_{i+1/2}} (qu - f)(x)\,dx = 0.$$

Вычислим $u'_{i\pm1/2}$ и интеграл, используя формулы (11), (12). Получим

$$-\frac{1}{h}\left(p_{i+1/2}\,\frac{u_{i+1}-u_i}{h} - p_{i-1/2}\,\frac{u_i - u_{i-1}}{h}\right) + q_i\,u_i = f_i + \psi_i, \qquad (13)$$

где $i = 2 : n-1$, ψ_i — погрешность аппроксимации, определяется как

$$\psi_i = \frac{h^2}{24}\,\frac{p_{i+1/2}\,u'''(\xi_{i+1/2}) - p_{i-1/2}\,u'''(\xi_{i-1/2})}{h} + \frac{h^2}{24}\,(f - qu)''(\xi_i). \quad (14)$$

Поскольку $f(x) - f(y) = f'(\xi)(x-y)$, $\xi \in [x, y]$, то

$$S = \frac{p_{i+1/2}\,u'''(\xi_{i+1/2}) - p_{i-1/2}\,u'''(\xi_{i-1/2})}{h} = \frac{p_{i+1/2} - p_{i-1/2}}{h}\,u'''(\xi_{i+1/2}) +$$
$$+ p_{i-1/2}\,\frac{u'''(\xi_{i+1/2}) - u'''(\xi_{i-1/2})}{h} = p'(\eta_{i-1/2})u'''(\xi_{i+1/2}) +$$
$$+ p_{i-1/2}u^{(4)}(\eta_{i+1/2})\,\frac{\xi_{i+1/2} - \xi_{i-1/2}}{h} \quad \Rightarrow \quad |S| \leqslant C = \text{const}.$$

Поэтому из (14) следует

$$|\psi_i| \leqslant C\,h^2, \quad i = 2 : n-1, \qquad (15)$$

где постоянная C не зависит от h.[1)]

2. Определение разностной схемы. Отбросим в (13) малую величину ψ_i и заменим $u(x)$ на приближение $y(x)$. Получим:

$$-\frac{1}{h}\left(p_{i+1/2}\,\frac{y_{i+1}-y_i}{h} - p_{i-1/2}\,\frac{y_i - y_{i-1}}{h}\right) + q_i\,y_i = f_i, \quad i = 2 : n-1, \quad (16)$$

[1)] Оценка (15) справедлива, если $q \in C^{(2)}[0,\ell]$, $f \in C^{(2)}[0,\ell]$, $u \in C^{(4)}[0,\ell]$. Из уравнения следует, что последнее верно, если также $p \in C^{(3)}[0,\ell]$.

которые дополним краевыми условиями $y_1 = y_n = 0$. Полученная СЛАУ и есть искомая конечно-разностная схема. Представим ее в виде системы

$$\begin{gathered} y_1 = 0, \\ \alpha_i y_{i-1} + \beta_i y_i + \alpha_{i+1} y_{i+1} = f_i, \quad i = 2 : n-1, \\ y_n = 0, \end{gathered}$$

где $\alpha_i = -p_{i-1/2}/h^2$, $\beta_i = q_i - \alpha_i - \alpha_{i+1}$. В матричном виде запишем ее как $Ay = f$. Эта СЛАУ с трехдиагональной матрицей A может быть решена за $O(n)$ flop методом прогонки.

3. Устойчивость схемы. Важное свойство разностной схемы устанавливается в следующей теореме.

Теорема 1. *Разностная схема* (16) *является устойчивой, т.е. найдется не зависящая от* h *постоянная* $C > 0$ *такая, что*

$$\|y\|_\infty \leqslant C\,\|f\|_\infty, \quad \|y\|_\infty = \max_{i=2:n-1} |y(x_i)|. \tag{17}$$

Доказательство. Положим $w_i = p_{i-1/2}\,(y_i - y_{i-1})/h$. Тогда уравнение (16) примет вид $-(w_{i+1} - w_i)/h + q_i y_i = f_i$. Умножим его на $h\,y_i$ просуммируем полученное по $i = 2 : n-1$. Получим

$$-\sum_{i=2}^{n-1}(w_{i+1} - w_i)y_i + \sum_{i=2}^{n-1} q_i\,y_i^2 = \sum_{i=2}^{n-1} h\,f_i\,y_i\,.$$

Преобразуем первое слагаемое, учитывая, что $y_1 = y = 0$:

$$\begin{aligned} -(w_3 - w_2)y_2 - (w_4 - w_3)y_3 - \ldots - (w_n - w_{n-1})y_{n-1} = \\ = w_2(y_2 - y_1) + w_3(y_3 - y_2) + \ldots + w_n(y_n - y_{n-1}). \end{aligned}$$

В результате придем к равенству

$$\sum_{i=2}^{n} h\,p_{i-1/2}\left(\frac{y_i - y_{i-1}}{h}\right)^2 + \sum_{i=2}^{n-1} q_i\,y_i^2 = \sum_{i=2}^{n-1} h\,f_i\,y_i\,.$$

Оценка правой части есть $\ell\,\|f\|_\infty\|y\|_\infty$. Учтем условия (3). Получим

$$\alpha\sum_{i=2}^{n} h\left(\frac{y_i - y_{i-1}}{h}\right)^2 \leqslant \ell\,\|f\|_\infty\|y\|_\infty\,. \tag{18}$$

Пусть $|y_j| = \|y\|_\infty$. Тогда

$$y_j = \sum_{i=2}^{j} h\,\frac{y_i - y_{i-1}}{h}.$$

Возведем это равенство в квадрат и используем неравенство Коши–Буняковского $\Big(\sum_{i=2}^{j} h w_i\Big)^2 \leqslant \Big(\sum_{i=2}^{j} h\Big)\Big(\sum_{i=2}^{j} h w_i^2\Big)$. Получим

$$\|y\|_\infty^2 = |y_j|^2 \leqslant \ell \sum_{i=2}^{n} h \left(\frac{y_i - y_{i-1}}{h}\right)^2.$$

Учтем эту оценку в (18). Тогда придем к (17) при $C = \ell/\sqrt{\alpha}$. $\square$

Следствие 1. *Решение разностной схемы существует и единственно.*

Доказательство. Разностная схема есть СЛАУ $Ay = f$. Однородной СЛАУ соответствует $f = 0$ и ее решение есть нулевой вектор. Это следует из оценки устойчивости (17). $\square$

Замечание 1. Поясним, почему оценка (17) называется оценкой устойчивости. Пусть y и $\bar{y}$ есть решения двух разностных схем с правыми частями f и $\bar{f}$, т.е. $Ay = f$, $A\bar{y} = \bar{f}$. Тогда $A(y - \bar{y}) = f - \bar{f}$ и $\|y - \bar{y}\|_\infty \leqslant C\,\|f - \bar{f}\|_\infty$ в силу оценки (17). Следовательно, если $\|f - \bar{f}\|_\infty \leqslant \varepsilon$, то $\|y - \bar{y}\|_\infty \leqslant C\,\varepsilon$, т.е. решение разностной схемы устойчиво к возмущению правой части.

4. Оценка точности схемы. Из устойчивости схемы и оценки ее погрешности аппроксимации следует оценка точности.

Теорема 2. *Разностная схема* (16) *имеет второй порядок точности, т.е. найдется не зависящая от h постоянная C такая, что*

$$\|u - y\|_\infty \leqslant C\,h^2. \tag{19}$$

Доказательство. Положим $z_i = u_i - y_i$. Вычтем из равенства (13) почленно равенство (16). Получим: $z_1 = z_n = 0$,

$$-\frac{1}{h}\left(p_{i+1/2}\,\frac{z_{i+1} - z_i}{h} - p_{i-1/2}\,\frac{z_i - z_{i-1}}{h}\right) + q_i\,z_i = \psi_i, \quad i = 2 : n-1.$$

Эта схема только правой частью отличается от исходной. В силу оценки (17) получим $\|z\|_\infty \leqslant C\,\|\psi\|_\infty$. Отсюда следует (19) в силу (15).

Замечание 2. Многие другие способы построения разностных схем также приводят к СЛАУ с трехдиагональной матрицей. Однако коэффициенты разностной схемы могут иметь другой вид, чем в (16). Например, коэффициент $a_i = p_{i-1/2}$ можно заменить на

$$a_i = \frac{p_i + p_{i-1}}{2} \quad \text{или} \quad a_i = \Big[\frac{1}{h}\int_{x_{i-1}}^{x_i} \frac{1}{p(x)}\,dx\Big]^{-1} \quad \text{или} \quad a_i = \frac{1}{h}\int_{x_{i-1}}^{x_i} p(x)dx,$$

или их аппроксимации с точностью $O(h^2)$. При этом оценка (15), а также оценка точности (19), сохранятся.

§ 3. Метод Галеркина

Для определения семейства методов Галеркина недостаточно формулировки задачи в виде уравнений (1), (2). Нужно связанное с ними интегральное тождество специального вида. Получим его.

Определение 1. $\mathcal{H}^1[0,\ell]$ *есть множество непрерывных кусочно-непрерывно дифференцируемых функций на отрезке* $[0,\ell]$.

Отметим, что $C^1[0,\ell] \subset \mathcal{H}^1[0,\ell]$. Далее для функций $v \in \mathcal{H}^1[0,\ell]$ под $v'(x)$ *будем понимать* определенную на $[0,\ell]$ функцию, совпадающую с кусочной производной $v(x)$. Для дифференцируемых функций $v(x)$ такая производная совпадает с обычной производной; в общем случае она будет разрывной функцией. По определению не важно, какое значение приписывается производной в точке разрыва. Так, функция $v = |x| \in \mathcal{H}^1[-1,1]$; ее кусочная производная $v'(x) = -1$ на отрезке $[-1,0]$, $v'(x) = 1$ на отрезке $[0,1]$.

Напомним *формулу интегрирования по частям*

$$\int_0^\ell v'(x)w(x)\,dx = -\int_0^\ell v(x)w'(x)\,dx + vw|_0^\ell, \tag{20}$$

которая справедлива для любых функций $v, w \in C^1[0,\ell]$.

Лемма 1. *Для функций* $v, w \in \mathcal{H}^1[0,\ell]$ *также справедлива формула интегрирования по частям* (20).

Доказательство. Пусть $v, w \in \mathcal{H}^1[0,\ell]$, $0 = a_1 < a_2 < \ldots < a_{K+1} = \ell$ и $v, w \in C^1[a_k, a_{k+1}]$ для всех k. Тогда, пользуясь (20) на отрезках

гладкости функций $v(x)$ и $w(x)$, получим:

$$\int_0^\ell v'(x)w(x)\,dx = \sum_{k=1}^{K}\int_{a_k}^{a_{k+1}} v'(x)w(x)\,dx = -\sum_{k=1}^{K}\int_{a_k}^{a_{k+1}} v(x)w'(x)\,dx + \\ + \sum_{k=1}^{K} vw\Big|_{a_k}^{a_{k+1}} = -\int_0^\ell v(x)w'(x)\,dx + vw\Big|_0^\ell. \qquad \square$$

Введем множество (пространство) функций

$$V = \{v \in \mathcal{H}^1[0,\ell] :\ v(0) = v(\ell) = 0\}. \tag{21}$$

Умножим уравнение (1) на $v \in V$ и проинтегрируем полученное равенство по отрезку $[0,\ell]$. Тогда придем к равенству

$$-\int_0^\ell (pu')'v\,dx + \int_0^\ell quv\,dx = \int_0^\ell fv\,dx \quad \forall\, v \in V. \tag{22}$$

Поскольку $pu' \in C^1[0,\ell]$, то можно воспользоваться формулой интегрирования по частям для преобразования интеграла, содержащего производные функции $u(x)$. Внеынтегральные слагаемые при этом будут равны нулю в силу выбора $v(x)$. В результате получим, что

$$\int_0^\ell (pu'v' + quv)\ dx = \int_0^\ell fv\,dx \quad \forall\, v \in V. \tag{23}$$

Соотношение (23) принято называть интегральным тождеством, соответствующим краевой задаче (1), (2). Его часто используют при построении методов приближенного решения задач вида (1), (2).

1. Метод Галеркина. Введем в рассмотрение линейно-независимые (базисные) функции

$$\varphi_1(x), \varphi_2(x), \ldots, \varphi_n(x) \in V. \tag{24}$$

Множество их линейных комбинаций обозначим через V_n, т.е.

$$V_n = \{v :\ v(x) = \sum_{i=1}^{n} c_i\,\varphi_i(x),\ c_i \in \mathbb{R}\}.$$

Можно поступить также следующим образом: определить конечномерное пространство функций $V_n \subset V$, а в нем выбрать базис (24).

Определение 2. *Приближенным решением задачи* (1), (2) *по методу Галеркина называется функция* $u_n \in V_n$ *такая, что*

$$\int_0^\ell (pu_n'v_n' + qu_nv_n)\,dx = \int_0^\ell fv_n\,dx \quad \forall\, v_n \in V_n. \tag{25}$$

Говорят, что тождество (25) определяет схему (метод) Галеркина для исходной задачи. Легко видеть, что равенства

$$\int_0^\ell (pu_n'\varphi_i' + qu_n\varphi_i)\,dx = \int_0^\ell f\varphi_i\,dx, \quad i = 1:n, \tag{26}$$

равносильны (25). В самом деле, выбирая в (25) $v_n = \varphi_i$ из (25) получим (26). Обратно, умножая (26) на произвольные числа c_i, после суммирования получим (25) для произвольной $v_n = \sum_{i=1}^n c_i\varphi_i \in V_n$.

Будем разыскивать приближенное решение в виде разложения

$$u_n(x) = \sum_{j=1}^n c_j\varphi_j(x).$$

После его подстановки в (26), для определения коэффициентов $c_1, \dots c_n$ получим систему линейных алгебраических уравнений:

$$\sum_{j=1}^n \Big(\int_0^\ell (p\varphi_j'\varphi_i' + q\varphi_j\varphi_i)\,dx \Big)\, c_j = \int_0^\ell f\varphi_i\,dx, \quad i = 1:n. \tag{27}$$

Запишем эту систему в матричном виде как $Ac = F$. Для ее формирования требуется вычислять интегралы (элементы A и F) вида

$$a_{ij} = \int_0^\ell (p\varphi_j'\varphi_i' + q\varphi_j\varphi_i)\,dx, \quad F_i = \int_0^\ell f\varphi_i\,dx. \tag{28}$$

Замечание 3. Свойства матрицы системы (27) и точность приближенного решения u_n определяются выбором базисной системы $\varphi_1, \dots, \varphi_n$. Если она выбрана удачно, то с увеличением n точность приближенного метода должна улучшаться. На практике для вычисления интегралов (28), обычно, применяют квадратурные формулы.

Лемма 2. *Матрица* A *системы линейных уравнений* (27) *симметрична и положительно определена. Как следствие, система* (27) *однозначно разрешима.*

Доказательство. По определению $a_{ij} = a_{ji}$. Напомним, что матрица A является положительно определенной, если ее квадратичная форма

$Ac \cdot c = \sum_{i,j=1}^{n} a_{ij} c_j c_i$ положительна для всех ненулевых векторов $c = (c_1, \ldots, c_n)^T$. Пусть c такой вектор-столбец, $v_n = \sum_{i=1}^{n} c_i \varphi_i$. Тогда v_n отлична от нулевой функции. Пользуясь (28), будем иметь:

$$Ac \cdot c = \sum_{i,j=1}^{n} a_{ij}\ c_j\, c_i = \int_0^\ell (p|v_n'|^2 + q|v_n|^2)\, dx > 0.$$

Поскольку определитель положительно определенной матрицы больше нуля, то система уравнений $Ac = F$ однозначно разрешима при любой правой части F. $\square$

В качестве примеров базисных систем метода Галеркина укажем:

a) полиномиальную систему:

$$\varphi_1(x) = \omega(x),\ \varphi_1(x) = \omega(x)\, x,\ \ldots, \varphi_n(x) = \omega(x) x^{n-1},$$

где $\omega(x) = x(\ell - x)$. В этом случае V_n есть подмножество множества алгебраических полиномов степени не выше $n + 1$.

b) полиномиальную систему:

$$\varphi_1(x) = \rho(x)\phi_0(x),\ \varphi_1(x) = \rho(x)\, \phi_1(x),\ \ldots, \varphi_n(x) = \rho(x)\phi_{n-1}(x),$$

где $\rho(x) = x(\ell - x)$, $\{\phi_k(x) = P_k^{(1,1)}(x)\}_{k=0}^{n-1}$ — система ортогональных с весом ρ полиномов Якоби на $[0, \ell]$. В этом случае V_n то же, что и в примере a), но в нем выбран другой базис.

c) тригонометрическую систему:

$$\varphi_k(x) = \sin\left(\frac{k\pi x}{\ell}\right), \quad k = 1, 2, \ldots, n. \tag{29}$$

В этом случае V_n есть подмножество множества тригонометрических полиномов степени не выше n.

При использовании указанных (или аналогичных) базисных систем матрица метода Галеркина оказывается заполненной, то есть все ее элементы отличны от нуля. Это принципиально отличает метод Галеркина от конечно-разностного метода, при использовании которого матрица системы линейных уравнений — разреженная матрица.

Оказывается, что при специальном выборе базисных функций метод Галеркина также приводит к системам линейных уравнений с разреженными матрицами. В настоящее время подобные методы, называемые методами конечных элементов (МКЭ), принадлежат к числу

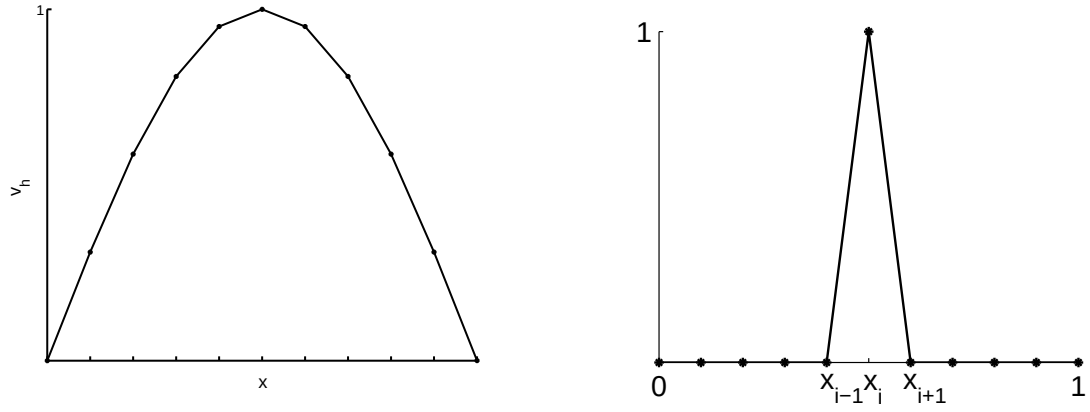

Рис. 1. Кусочно линейная функция $u_h(x)$ (слева). Базисная функция $\varphi_i(x)$ (справа).

наиболее распространенных методов решения эллиптических уравнений и систем. Опишем вариант МКЭ на примере задачи (1), (2).

§ 4. Метод конечных элементов

Укажем выбор пространства V_n в методе Галеркина, далее обозначаемый через V_h, приводящий к МКЭ. Для этого определим на отрезке $[0, \ell]$ сетку узлов

$$\omega_h = \{0 = x_0 < x_1 < \cdots < x_{n+1} = \ell\}\,.$$

Ячейки сетки $e_i = [x_{i-1},\ x_i]$, $i = 1 : n+1$, длины h_i, назовем конечными элементами. Положим $h = \max_{1 \leqslant i \leqslant n+1} h_i$.

Пространство V_h определим как множество непрерывных на отрезке $[0, \ell]$ функций $u_h(x)$, равных нулю при $x = 0$ и $x = \ell$ и линейных на каждом элементе e_i (см. левый рис. 1). Нетрудно видеть, что

$$u_h(x) = \sum_{i=1}^{n} c_i \varphi_i(x), \quad x \in [0, \ell], \tag{30}$$

где $c_i = u_h(x_i)$, $\varphi_i(x)$ — линейные на каждом элементе, непрерывные на $[0, \ell]$ функции, удовлетворяющие условиям (см. правый рис. 1):

$$\varphi_i(x_j) = \begin{cases} 1, & i = j, \\ 0, & i \neq j, \quad j = 1, \ldots, n. \end{cases}$$

Определение 3. *Приближенным решением задачи* (1), (2) *по МКЭ на основе линейных элементов называется функция* $u_h \in V_h$, *удовлетворяющая для любого* $v_h \in V_h$ *равенству*

$$\int_0^\ell (pu_h'v_h' + qu_hv_h)\,dx = \int_0^\ell fv_h\,dx. \tag{31}$$

Говорят, что тождество (31) определяет схему МКЭ на основе линейных элементов для исходной задачи.

Так как МКЭ есть вариант метода Галеркина, то задача (31) равносильна СЛАУ $Ac = F$ для определения коэффициентов в разложении (30), причем элементы A и F имеют вид

$$a_{ij} = \int_0^\ell (p\varphi_j'\varphi_i' + q\varphi_j\varphi_i)\,dx, \quad F_i = \int_0^\ell f\varphi_i\,dx.$$

Нетрудно видеть, что $a_{ij} = 0$, если $|i-j| \geqslant 2$, т.к. в этом случае $\varphi_j'\varphi_i' = \varphi_j\varphi_i \equiv 0$ на $[0,\ell]$.

Пример. Рассмотрим равномерную сетку с шагом $h = \ell/(n+1)$ и пусть $p = \text{const}$, $q = \text{const}$. Матрицу A представим в виде суммы двух матриц K и M, $A = K + M$, где

$$K = \left\{k_{ij} = p\int_0^\ell \varphi_j'\varphi_i'\,dx\right\}_{i,j=1}^n, \quad M = \left\{m_{ij} = q\int_0^\ell \varphi_j\varphi_i\,dx\right\}_{i,j=1}^n.$$

Нетрудно вычислить (неуказанные элементы матриц равны нулю):

$$K = \frac{p}{h}\begin{pmatrix} 2 & -1 & & & \\ -1 & 2 & -1 & & \\ & & \dots & & \\ & & -1 & 2 & -1 \\ & & & -1 & 2 \end{pmatrix}, \quad M = \frac{q\,h}{6}\begin{pmatrix} 4 & 1 & & & \\ 1 & 4 & 1 & & \\ & & \dots & & \\ & & 1 & 4 & 1 \\ & & & 1 & 4 \end{pmatrix}.$$

Вычислим элементы правой части, считая, что $f(x) \approx f(x_i)$ на отрезке $[x_{i-1}, x_{i+1}]$. Тогда

$$F_i = \int_{x_{i-1}}^{x_{i+1}} f(x)\varphi_i(x)\,dx \approx f(x_i)\int_{x_{i-1}}^{x_{i+1}} \varphi_i(x)\,dx = h\,f(x_i).$$

В этом случае, система $Ac = F$ в индексном виде примет вид

$$-p\,\frac{c_{i-1} - 2c_i + c_{i+1}}{h} + \frac{q\,h}{6}\,(c_{i-1} + 4c_i + c_{i+1}) = h\,f(x_i), \quad i = 1:n.$$

К этим уравнениям надо присоединить краевые условия

$$c_0 = 0,\ c_{n+1} = 0.$$

Замечание 4. Другие варианты МКЭ, более высокой точности, получаются, если пространство V_h определить не на основе лагранжевых сплайнов первой степени S_h^1, а на основе лагранжевых сплайнов S_h^m — степени m. В этом случае

$$V_h = \{v_h \in S_h^m : v_h(0) = v_h(\ell) = 0\}.$$

Задания для самостоятельной работы

ВОПРОСЫ ДЛЯ САМОКОНТРОЛЯ

1. Как определяется метод коллокации для решения дифференциальных уравнений?

2. Запишите систему алгебраических уравнений метода коллокации для ОДУ второго порядка. Дайте определение матрицы дифференцирования а) первого порядка. б) k-го порядка.

3. В чем состоит основная идея метода конечных разностей для решения дифференциальных уравнений?

4. Приведите: а) формулу центральной разности для вычисления производной первого порядка функции. б) квадратурную формулу центральных прямоугольников для вычисления интеграла. с) приведите оцени погрешности этих формул.

5. а) Запишите разностную схему для решения задачи

$$-u''(x) + u(x) = f(x), \quad 0 < x < 1, \quad u(0) = \alpha, \ u(1) = \beta,$$

на равномерной сетке узлов. б) Приведите оценку точности этой схемы.

6. За сколько арифметических операций можно найти решение разностной схемы для ОДУ 2-го порядка и каким методом?

7. а) Приведите схему МКЭ для решения задачи

$$-u''(x) + u(x) = f(x), \quad 0 < x < 1, \quad u(0) = \alpha, \ u(1) = \beta,$$

на равномерной сетке узлов. Сравните ее с разностной схемой. В чем отличия этих схем в данном случае?

8. За сколько арифметических операций можно найти решение разностной схемы и каким методом?

ЗАДАЧИ И УПРАЖНЕНИЯ

1. а) Найдите решение краевой задачи

$$-u''(x) = 2, \quad 0 < x < 1, \quad u(0) = u(1) = 0.$$

б) запишите конечно-разностную схему ее решения на сетке с 4 равномерно расположенными узлами. Найдите ее решение и погрешность;

с) постройте график точного и решения и решения разностной схемы;

d) объясните полученный результат, оценив предварительно погрешность аппроксимации схемы.

2. а) Для решения краевой задачи

$$-u''(x) = 2, \quad 0 < x < 1, \quad u(0) = u(1) = 0.$$

постройте схему Галеркина с двумя полиномиальными базисными функциями.

б) Найдите решения этой схемы и ее погрешность.

с) Как можно объяснить полученный результат?

3. а) Для решения краевой задачи

$$-u''(x) = 2, \quad 0 < x < 1, \quad u(0) = u(1) = 0.$$

постройте схему МКЭ на сетке с 4 равномерно расположенными узлами.

б) Найдите решения этой схемы и ее погрешность.

с) Постройте график точного и решения и решения схемы МКЭ.

d) Объясните полученный результат.

5. а) Запишите разностную схему для решения задачи

$$-u''(x) + (1 + x)\, u(x) = f(x), \quad 0 < x < 1, \quad u(0) = \alpha, \ u(1) = \beta,$$

на равномерной сетке узлов. б) Исследуйте ее разрешимость.

ГЛАВА 7

Решение уравнений в частных производных

§ 1. Разностные методы для уравнения теплопроводности

Рассмотрим задачу Дирихле для уравнения теплопроводности

$$\frac{\partial u}{\partial t} - \frac{\partial}{\partial x}\left(p\,(x)\,\frac{\partial u}{\partial x}\right) + q(x)\,u = f(x,t), \quad 0 < x < \ell,\ t \in (0,T], \quad (1)$$

$$u(x,0) = u_0(x), \quad 0 < x < \ell, \quad (2)$$

$$u(0,t) = u(\ell,t) = 0, \quad t \in (0,T], \quad (3)$$

для определения неизвестной функции $u = u(x,t)$.

Как и ранее предполагаем, что выполнены условия

$$0 < \alpha \leqslant p\,(x) \leqslant \beta, \quad 0 \leqslant q(x) \leqslant \gamma, \quad \alpha, \beta, \gamma = \text{const}.$$

1. Метод прямых (метод полудискретизации по x). Рассмотрим равномерную на $[0,\ell]$ сетку узлов

$$\omega_h = \{x_i = (i-1)h,\ i = 1:n,\ h = \ell/(n-1)\}\,.$$

В методе прямых ищутся приближения $y_i(t) = y(x_i,t)$ к решению $u_i(t) = u(x_i,t)$ в каждой точке сетки x_i при $t \in [0,T]$. Поскольку в силу краевых условий естественно принять $y_1(t) = y(0,t) = 0$, $y_n(t) = y(\ell,t) = 0$, то необходимо указать способ определения вектор функции неизвестных

$$y(t) = (y_2(t), \ldots, y_{n-1}(t))^T = (y(x_2,t), \ldots, y(x_{n-1},t))^T, \quad t \in [0,T]. \quad (4)$$

Для определения этого вектора строится задача Коши вида

$$y'(t) + Ay(t) = f(t), \quad y(0) = u_0, \quad (5)$$

которая может быть решена, например, явным или неявным методом Эйлера (или любым другим методом).

Определим *конечно-разностный метод* вывода схемы (5). Рассмотрим уравнение (1) при $x = x_i$, $i = 2 : n-1$. По определению

$$u_i'(t) = \frac{\partial}{\partial t}\, u(x_i, t).$$

Положим $F(x,t) = f(x,t) - u_i'(t)$. Тогда уравнение (1) примет вид

$$-\frac{\partial}{\partial x}\left(p\,(x)\,\frac{\partial u}{\partial x}\right) + q(x)\,u = F(x,t), \quad 0 < x < \ell.$$

В это уравнение переменная $t \in (0,T]$ входит как параметр. Аппроксимируем его в точке сетки x_i методом баланса. Получим

$$-\frac{1}{h}\left(p_{i+1/2}\,\frac{u_{i+1}-u_i}{h} - p_{i-1/2}\,\frac{u_i - u_{i-1}}{h}\right) + q_i\,u_i = F_i + \psi_i, \qquad (6)$$

где $i = 2 : n-1$, и для краткости принято $u_i = u_i(t)$, $F_i = F(x_i,t)$. Для погрешности аппроксимации $\psi_i = \psi_i(t)$ нами была получена оценка

$$|\psi_i(t)| \leqslant C\,h^2, \quad i = 2 : n-1, \qquad (7)$$

где постоянная C не зависит от h.[1)] Равенства (6) запишем в виде

$$u_i' + \alpha_{\,i} u_{i-1} + \beta_i\,u_i + \alpha_{i+1}\,u_{i+1} = f_i + \psi_i, \quad i = 2 : n-1,$$
$$u_1 = u_n = 0,$$

где $\alpha_i = -p_{i-1/2}/h^2$, $\beta_i = q_i - \alpha_i - \alpha_{i+1}$. Из этой системы исключим u_1 и u_n. Тогда придем к системе ОДУ первого порядка вида

$$u'(t) + Au(t) = f(t) + \psi(t), \quad t \in (0,T], \qquad (8)$$

где $u(t) = (u_2(t), \ldots, u_{n-1}(t))^T$, матрица A и функция f имеют вид

$$A = \begin{pmatrix} \beta_2 & \alpha_3 & & & \\ \alpha_3 & \beta_3 & \alpha_4 & & \\ & & \ldots & & \\ & & \alpha_{n-2} & \beta_{n-2} & \alpha_{n-1} \\ & & & \alpha_{n-1} & \beta_{n-1} \end{pmatrix}, \quad f(t) = \begin{pmatrix} f(x_2,t) \\ f(x_3,t) \\ \ldots \\ f(x_{n-2},t) \\ f(x_{n-1},t) \end{pmatrix}.$$

Отбрасим в (8) малую величину $\psi(t) = O(h^2)$, и заменим u на y. Придем к системе ОДУ

$$y'(t) = \Phi(t, y(t)), \quad t \in (0,T], \quad \Phi(t,y) = f(t) - Ay, \qquad (9)$$

[1)] Оценка (7) справедлива, если $q \in C^{(2)}[0,\ell]$, и $F \in C^{(2)}[0,\ell]$, $u \in C^{(4)}[0,\ell]$ при каждом $t \in [0,T]$.

которое дополним начальным условием

$$y(0) = u_0, \quad u_0 = (u_0(x_2), \ldots, u_0(x_{n-1}))^T. \tag{10}$$

Задача Коши (9), (10) определяет метод прямых для приближенного решения исходной задачи в частных производных.

Таким образом, в этом методе задача в частных производных аппроксимируется более простой — задачей Коши для системы ОДУ. Отметим, что число уравнений в этой системе равно числу узлов сетки и может быть велико. Для решения задачи Коши можно использовать один из рассмотренных нами методов. В связи с этим изучим два простейших метода.

2. Явная разностная схема. Используем явный метод Эйлера для решения задачи Коши (9), (10). В результате придем к явной разностной схеме решения исходной задачи в частных производных.

Для этого определим на $[0, T]$ равномерную сетку с шагом τ:

$$\omega_\tau = \{t_j = j\tau,\ j = 0 : M,\ \tau = T/M\}.$$

Значение функции f в точке t_j будем обозначать через f^j. Так что $f^j = f(t_j)$. Согласно (4) получаем

$$y^j = (y_2^j, \ldots, y_{n-1}^j)^T = (y(x_2, t_j), \ldots, y(x_{n-1}, t_j))^T.$$

Аппроксимируем равенство (8), используя формулу численного дифференцирования первого порядка аппроксимации:

$$u'(t) = \frac{u(t+\tau) - u(t)}{\tau} + \mu(t), \quad \mu(t) = -\frac{\tau}{2}\, u''(\xi(t)) = O(\tau).$$

Рассматривая равенство (8) при $t = t_j$ получим:

$$\frac{u^{j+1} - u^j}{\tau} + Au^j = f^j + \psi^j + \mu^j, \quad j = 0 : M-1.$$

Отбрасывая здесь величины $\psi^j + \mu^j = O(\tau + h^2)$, и, заменяя u на y, придем к явной разностной схеме:

$$y^{j+1} = y^j + \tau\,(f^j - Ay^j), \quad j = 0 : M-1, \quad y^0 = u_0. \tag{11}$$

Если матрицу A хранить в памяти ЭВМ в разреженном формате, то вычисления по формулам (11) являются экономичными и позволяют за $O(nM)$ flop вычислить все y^j, $j = 0 : M$. Можно воспользоваться также индексной записью системы уравнений (11):

$$y_i^{j+1} = d_i\, y_i^j + \frac{\tau\, p_{i-1/2}}{h^2}\, y_{i-1}^j + \frac{\tau\, p_{i+1/2}}{h^2}\, y_{i+1}^j + \tau\, f_i^j, \quad i = 2 : n-1, \tag{12}$$
$$y_1^{j+1} = y_n^{j+1} = 0,$$

где $d_i = 1 - \tau\,(p_{i-1/2} + p_{i+1/2})/h^2 - \tau\, q_i$.

Описанный метод дает возможность вычислить приближение к решению $u(x,t)$ в моменты времени t_{j+1}, $j = 0 : M-1$, в виде таблицы

x_1	x_2	$\dots$	x_{n-1}	x_n
0	y_2^{j+1}	$\dots$	y_{n-1}^{j+1}	0

3. Неявная разностная схема. Используя формулу численного дифференцирования

$$u'(t+\tau) = \frac{u(t+\tau) - u(t)}{\tau} + \mu(t+\tau), \quad \mu(t) = \frac{\tau}{2}\, u''(\xi(t)) = O(\tau),$$

рассмотрим (8) при $t = t_{j+1}$. Аналогично предыдущему получим:

$$\frac{u^{j+1} - u^j}{\tau} + Au^{j+1} = f^{j+1} + \Psi^{j+1}, \quad j = 0 : M-1, \tag{13}$$

где $\Psi^{j+1} = \psi^{j+1} + \mu^{j+1}$ — погрешность аппроксимации, $\Psi^{j+1} = O(\tau + h^2)$. Отбросим здесь Ψ^{j+1} и, заменим u на y. Придем к неявной схеме:

$$y^{j+1} + \tau\, Ay^{j+1} = y^j + \tau\, f^{j+1}, \quad j = 0 : M-1, \quad y^0 = u_0. \tag{14}$$

В отличие от явной схемы, при каждом j мы должны определять y^{j+1} из решения СЛАУ с трехдиагональной матрицей $I + \tau\, A$. Методом прогонки это можно сделать за $O(n)$ flop.

Запишем систему систему уравнений (14) в индексной форме:

$$-A_i y_{i-1}^{j+1} + B_i y_i^{j+1} - A_{i+1} y_{i+1}^{j+1} = y_i^j + \tau\, f_i^{j+1}, \quad i = 2 : n-1, \tag{15}$$
$$y_1^{j+1} = y_n^{j+1} = 0,$$

где

$$A_i = \frac{\tau p_{i-1/2}}{h^2}, \quad B_i = 1 + A_i + A_{i+1} + \tau\, q_i. \tag{16}$$

Как отмечалось при изучении методов решения задачи Коши для ОДУ, разница между явными и неявными методами кроется в их устойчивости. Проясним этот вопрос для рассмотренных разностных схем.

4. Устойчивость и точность неявной схемы. Введем обозначения для норм:

$$|y^j|_\infty = \max_{2\leqslant i\leqslant n-1} |y(x_i,t_j)|, \quad \|y\|_\infty = \max_{1\leqslant j\leqslant M} |y^j|_\infty .$$

Теорема 1. *Пусть y — решение неявной схемы* (14). *Тогда справедлива следующая оценка устойчивости:*

$$\|y\|_\infty \leqslant |u_0|_\infty + T\,\|f\|_\infty .$$

Доказательство. Определим i, $i = 2 : n-1$, равенством

$$|y_i^{j+1}| = |y^{j+1}|_\infty, \tag{17}$$

и рассмотрим уравнение (15), которое перепишем в следующем виде:

$$B_i\, y_i^{j+1} = A_i\, y_{i-1}^{j+1} + A_{i+1}\, y_{i+1}^{j+1} + y_i^j + \tau f_i^{j+1} .$$

Отсюда вытекает очевидная оценка

$$B_i\, |y_i^{j+1}| \leqslant A_i\, |y_{i-1}^{j+1}| + A_{i+1}\, |y_{i+1}^{j+1}| + |y_i^j| + \tau\, |f_i^{j+1}| .$$

Следовательно,

$$B_i\, |y^{j+1}|_\infty \leqslant (A_i + A_{i+1})\, |y^{j+1}|_\infty + |y^j|_\infty + \tau\, |f^{j+1}|_\infty .$$

Учитывая, что $B_i - A_i - A_{i+1} = 1 + \tau\, q_i \geqslant 1$, получаем

$$|y^{j+1}|_\infty \leqslant |y^j|_\infty + \tau\,\|f\|_\infty, \quad j = 0 : M-1. \tag{18}$$

Воспользуемся неравенством (18) рекуррентно. Тогда

$$\begin{aligned} |y^j|_\infty \leqslant |y^{j-1}|_\infty + \tau\,\|f\|_\infty \leqslant |y^{j-2}|_\infty + 2\tau\,\|f\|_\infty \leqslant \ldots \leqslant \\ \leqslant |y^0|_\infty + j\,\tau\,\|f\|_\infty \leqslant |u_0|_\infty + T\,\|f\|_\infty . \end{aligned}$$ □

Теорема 2. *Пусть y — решение неявной схемы* (14). *Тогда*

$$\|u - y\|_\infty = O(\tau + h^2). \tag{19}$$

Доказательство. Из соотношений (13) и (14) следует, что погрешность $z = u - y$ является решением неявной схемы, но с правой частью, равной погрешности аппроксимации. А именно,

$$z^{j+1} + \tau\, A z^{j+1} = z^j + \tau\, \Psi^{j+1}, \quad j = 0 : M-1, \quad y^0 = 0. \tag{20}$$

В силу оценки устойчивости имеем $\|z\|_\infty \leqslant T\,\|\Psi\|_\infty$. Отсюда следует (19), т.к. $\Psi^{j+1} = O(\tau + h^2)$. $\square$

Отметим, что величина $\|u-y\|_\infty$ равна максимальной абсолютной погрешности приближенного решения в точках сетки $\omega_h \times \omega_\tau$.

5. Устойчивость и точность явной разностной схемы. Теорема 3. *Пусть y — решение явной схемы* (11)*, шаги сетки удовлетворяют условию*

$$\tau \leqslant \tau_0 = \frac{h^2}{2\,|p\,|_\infty + h^2 |q|_\infty}, \quad |p\,|_\infty = \max_{x\in[0,\ell]} p\,(x). \tag{21}$$

Тогда справедлива оценка устойчивости:

$$\|y\|_\infty \leqslant |u_0|_\infty + T\,\|f\|_\infty\,. \tag{22}$$

Доказательство. Согласно определению явной схемы (см. (12))

$$y_i^{j+1} = d_i\, y_i^j + \frac{\tau\, p_{i-1/2}}{h^2}\; y_{i-1}^j + \frac{\tau\, p_{i+1/2}}{h^2}\; y_{i+1}^j + \tau\, f_i^j, \tag{23}$$

где $d_i = 1 - \tau\,(p_{i-1/2} + p_{i+1/2} + h^2\, q_i)/h^2$, $y_1^{j+1} = y_n^{j+1} = 0$.

Легко видеть, что при условии (21) коэффициент $d_i > 0$, т.к.

$$0 < \tau\,(p_{i-1/2} + p_{i+1/2} + h^2\, q_i)/h^2 \leqslant \tau_0\,(2|p\,|_\infty + h^2\,|q|_\infty)/h^2 = 1\,.$$

Поэтому при условии (21) из (23) получаем для $i = 2 : n-1$:

$$\begin{aligned} |y_i^{j+1}| &\leqslant d_i\,|y_i^j| + \frac{\tau\, p_{i-1/2}}{h^2}\;|y_{i-1}^j| + \frac{\tau\, p_{i+1/2}}{h^2}\;|y_{i+1}^j| + \tau\,|f_i^j| \leqslant \\ &\leqslant \Big(d_i + \frac{\tau\, p_{i-1/2}}{h^2} + \frac{\tau\, p_{i+1/2}}{h^2}\Big)|y^j|_\infty + \tau\,\|f\|_\infty = \\ &= (1 - \tau\, q_i)\,|y^j|_\infty + \tau\,\|f\|_\infty \leqslant |y^j|_\infty + \tau\,\|f\|_\infty. \end{aligned}$$

Эта оценка справедлива для любого $i = 2 : n-1$. Следовательно,

$$|y^{j+1}|_\infty \leqslant |y^j|_\infty + \tau\,\|f\|_\infty, \quad j = 0 : M-1. \tag{24}$$

Из оценки (24), как и в случае неявной схемы, выводится искомая оценка устойчивости (22). $\square$

Как и в случае неявной схемы выводится также

Теорема 4. *Пусть y — решение явной схемы* (14). *Тогда*

$$\|u - y\|_\infty = O(\tau + h^2).$$

Замечание 1. Устойчивость явной разностной схемы была нами доказана лишь при условии, что шаги сетки удовлетворяют неравенству (21). Это условие на шаги сетки, как можно доказать, является необходимым. При его нарушении решение явной разностной схемы быстро «разбалтывается». Условие (21) на практике оказывается особенно обременительным, если коэффициент p быстро меняется. Тогда шаг по времени приходится брать слишком маленьким, возможно существенно меньшим, чем это диктуется соображениями точности. Это ведет к неоправданному увеличению вычислительной работы. Неявная схема устойчива, как мы показали, при любых шагах сетки (абсолютно устойчива). При ее использовании шаги сетки можно выбирать лишь из соображений точности.

§ 2. Разностная схема для уравнения Пуассона

Рассмотрим метод конечных разностей для приближенного решения следующей задачи Дирихле для модельного уравнения эллиптического типа в прямоугольной области $\Omega = (0, L) \times (0, L)$:

$$-\Delta u(x) = -\frac{\partial^2 u(x)}{\partial x_1^2} - \frac{\partial^2 u(x)}{\partial x_2^2} = f(x), \quad x \in \Omega, \tag{25}$$

$$u(x) = 0, \quad x \in \partial\Omega. \tag{26}$$

Здесь $x = (x_1, x_2)$; $\partial\Omega$ обозначает множество граничных точек Ω; $f(x)$ — заданная непрерывная функция. Неизвестную функцию $u(x)$ требуется определить.

На области Ω зададим дискретное множество точек, в которых будем определять приближенное решение задачи (определим сетку). Для этого отрезки $[0, L]$ на осях x_1 и x_2 разобьем на n равных частей; пусть $h = L/n$. Через точки деления проведем прямые, параллельные соответствующим осям. В результате пересечения этих прямых получатся узлы, которые и образуют сетку. Те узлы (ih, jh), которые лежат внутри Ω, назовем внутренними. Их совокупность обозначим

$$\omega_h = \{(ih, jh) : \ i, j = 1, 2, \ldots, n-1\}.$$

Множество узлов сетки, принадлежащих $\partial\Omega$, назовем граничными и обозначим через γ_h.

Дискретный аналог задачи (25), (26) построим следующим образом: уравнения рассмотрим в точках сетки, а вторые производные аппроксимируем разностными отношениями:

$$\frac{\partial^2 u(x)}{\partial x_1^2} = \frac{u(x_1-h,x_2)-2u(x_1,x_2)+u(x_1+h,x_2)}{h^2}+\psi_1(x),$$
$$\frac{\partial^2 u(x)}{\partial x_2^2} = \frac{u(x_1,x_2-h)-2u(x_1,x_2)+u(x_1,x_2+h)}{h^2}+\psi_2(x).$$

Получим

$$Au(x) = f(x)+\psi(x), \quad x\in\omega_h, \tag{27}$$

где

$$Au(x) = -\frac{u(x_1-h,x_2)-2u(x_1,x_2)+u(x_1+h,x_2)}{h^2}- \\ -\frac{u(x_1,x_2-h)-2u(x_1,x_2)+u(x_1,x_2+h)}{h^2},$$

а для погрешности аппроксимации справедливо представление

$$\psi(x) = -\frac{h^2}{12}\frac{\partial^4}{\partial x_1^4}u(\xi_1(x),x_2)-\frac{h^2}{12}\frac{\partial^4}{\partial x_2^4}u(x_1,\xi_2(x)) = O(h^2)$$

если $u\in C^4(\overline{\Omega})$. Отбросим в (27) малую ψ и заменим $u(x)$ на его приближения $y(x)$. В результате придем к конечно-разностной схеме

$$Ay(x) = f(x),\ x\in\omega_h, \quad y(x)=0,\ x\in\gamma_h. \tag{28}$$

Она является дискретным аналогом исходной задачи.

1. Матричное представление схемы. Запишем систему уравнений (28) в матричном виде. Ясно, что неизвестными являются лишь значения $y(x)$ в точках ω_h. Поскольку значения $y(x)$ в точках γ_h известны и равны нулю, то их нет необходимости включать в вектор неизвестных. Учитывая сказанное, уравнения (28) запишем в виде

$$-y(x_1,x_2-h)-y(x_1-h,x_2)+4\,y(x_1,x_2)- \\ -y(x_1+h,x_2)-y(x_1,x_2+h) = h^2 f(x), \quad x\in\omega_h, \tag{29}$$

считая, что слагаемое вида $y(x_1,x_2\pm h)$ или $y(x_1\pm h,x_2)$ в этом равенстве опущено, если соответствующий ему узел сетки $(x_1,x_2\pm h)$ или $(x_1\pm h,x_2)$ принадлежит γ_h. Отметим, что такой коррекции требуют

лишь уравнения, соответствующие приграничным узлам (т. е. узлам (ih, jh) при i или j равным 1 или $n-1$).

Чтобы определить вектор неизвестных, необходимо пронумеровать узлы сетки ω_h. Ясно, что это можно сделать многими способами. Выберем следующий способ: узлы ω_h пронумеруем слева-напрво и снизу-вверх, начиная с узла с координатой (h, h). А именно, примем, что узел (ih, jh) имеет номер l (т. е. $x_l = (ih, jh)$), если

$$l = (j-1)(n-1) + i, \quad i, j = 1 : n-1.$$

В такой нумерации уравнения (29) запишутся в виде

$$-y_{l-n+1} - y_{l-1} + 4\,y_l - y_{l+1} - y_{l+n-1} = h^2 f_l, \quad l = 1 : N, \tag{30}$$

где $N = (n-1)^2$, $f_l = f(x_l)$. Уравнения (30) нужно скорректировать, опуская соответствующие слагаемые, если узел x_l является приграничным.

Уравнения (30) в матричном виде примут вид $A_N y = F_N$, где l-тая компонента y равна y_l, l-тая компонента F_N равна $h^2 f_l$, а матрица A_N размера N имеет следующий блочно-трехдиагональный вид:

$$A_N = \begin{bmatrix} T & -I & & & \\ -I & T & -I & & \\ & \dots & \dots & \dots & \\ & & -I & T & -I \\ & & & -I & T \end{bmatrix},$$

где I — единичная матрица размера $n-1$, T — трехдиагональная матрица размера $n-1$ вида

$$T = \begin{bmatrix} 4 & -1 & & & \\ -1 & 4 & -1 & & \\ & \dots & \dots & \dots & \\ & & -1 & 4 & -1 \\ & & & -1 & 4 \end{bmatrix}.$$

Матрица A_N является симметричной и разреженной (подавляющее число ее элементов — нули, ненулевые элементы расположены лишь на пяти диагоналях). Она невырождена (см. упр. 1) и положительно определена.

Отметим, что система уравнений $A_N y = F_N$ может иметь большую размерность. Например, при $n \approx 10^3$ получаем $N \approx 10^6$. Для ее решения можно использовать, например, метод Холецкого (учитывающий разреженность матрицы) или итерационные методы.

2. Устойчивость и точность разностной схемы. Пусть $\bar{\omega}_h = \omega_h \cup \gamma_h$. Обозначим через H_h множество всех сеточных функций, определенных на сетке ω_h, с нормой

$$\|y\|_\infty = \max_{x \in \omega_h} |y(x)|.$$

Установим предварительно несколько свойств разностной схемы.

Лемма 1. *Пусть y сеточная функция, не являющаяся тождественно постоянной на $\bar{\omega}_h$, и пусть $Ay(x) \leqslant 0$ для всех $x \in \omega_h$. Тогда функция $y(x)$ не может достигать максимального значения во внутренних точках $x \in \omega_h$.*

Доказательство. Предположим обратное. Пусть существует внутренняя точка сетки $x \in \omega_h$ такая, что $y(x) \geqslant y(x')$ для любого $x' \in \bar{\omega}_h$. Имеем

$$0 \geqslant Ay(x) = \frac{y(x_1, x_2) - y(x_1 - h, x_2)}{h^2} + \frac{y(x_1, x_2) - y(x_1 + h, x_2)}{h^2} + \\ + \frac{y(x_1, x_2) - y(x_1, x_2 - h)}{h^2} + \frac{y(x_1, x_2) - y(x_1, x_2 + h)}{h^2}.$$

Поскольку $y(x) \geqslant y(x')$ для любого x', то дроби здесь неотрицательны и обращается в нуль лишь только в том случае, когда

$$y(x_1 - h, x_2) = y(x_1, x_2) = y(x_1 + h, x_2) = y(x_1, x_2 - h) = y(x_1, x_2 + h).$$

Поскольку функция $y(x)$ не постоянная, то, очевидно, найдется такая точка сетки x'', в которой $Ax(x'') < 0$. Полученное противоречие доказывает лемму. □

Лемма 2. *Пусть y сеточная функция, не являющаяся тождественно постоянной на $\bar{\omega}_h$, и пусть $Ay(x) \geqslant 0$ во всех внутренних точках $x \in \omega_h$. Тогда функция $y(x)$ не может достигать минимального значения во внутренних точках $x \in \omega_h$.*

Доказательство. Доказательство аналогично лемме 1. □

Лемма 3. *Пусть* $Ay(x) = 0$ *для всех* $x \in \omega_h$. *Тогда*

$$\max_{x\in\bar{\omega}_h} |y(x)| \leqslant \max_{x\in\gamma_h} |y(x)|. \tag{31}$$

Доказательство. Если $y(x)$ всюду на $\bar{\omega}_h$ равна константе, то неравенство (31) очевидно. Иначе, для $y(x)$ справедливы утверждения как леммы 1, так и леммы 2. Из них следует, что $y(x)$ достигает максимального и минимального значений на γ_h. То же самое справедливо для функции $-y(x)$. Другими словами, справедлива оценка (31). □

Следствие 1. *Разностная схема* (28) *однозначно разрешима.*

Теорема 5. *Разностная схема* (28) *является устойчивой, т.е. для любого* $f \in H_h$ *справедлива оценка* $\|y\|_\infty \leqslant C\,\|f\|_\infty$, *где* $C = L^2/2$.

Доказательство. Положим

$$w(x) = F\,(2L^2 - x_1^2 - x_2^2)/4, \quad F = \|f\|_\infty.$$

Нетрудно проверяется, что (см. упр. 2)

$$Aw(x) = F,\ x \in \omega_h, \quad w(x) \geqslant 0,\ x \in \gamma_h. \tag{32}$$

Вычитая равенства (32) и (28), для $z = w - y$ получим схему

$$Az(x) = \Phi(x),\ x \in \omega_h, \quad z(x) \geqslant 0,\ x \in \gamma_h,$$

где $\Phi(x) = F - f(x) \geqslant 0$ для любого $x \in \omega_h$, т.е. мы находимся в условиях леммы 2. Следовательно, функция $z(x)$ не может достигать минимального значения внутри области, то есть она достигает его на границе γ_h, где она неотрицательна. Следовательно, $z(x) = w(x) - y(x)$ неотрицательна во всех точках сетки. Аналогично устанавливается, что $w(x) + y(x) \geqslant 0$ во всех точках сетки. Таким образом, $-w(x) \leqslant y(x) \leqslant w(x)$, т.е.

$$|y(x)| \leqslant |w(x)| = |F\,(2L^2 - x_1^2 - x_2^2)/4| \leqslant L^2\,F/2 = L^2/2\,\|f\|_\infty.$$

Отсюда следует утверждение теоремы. □

Теорема 6. *Пусть правая часть* $f(x)$ *уравнения* (25) *такова, что решение* $u(x)$ *задачи* (25), (26) *принадлежит* $C^4(\overline{\Omega})$. *Тогда*

$$\|u - y\|_\infty \leqslant C\,h^2\|f\|_\infty,$$

где постоянная C не зависит от h.

Доказательство. Погрешность решения $z(x) = u(x) - y(x)$, $x \in \bar{\omega}_h$, является решением разностной схемы

$$Az(x) = \psi(x),\ x \in \omega_h, \quad z(x) = 0,\ x \in \gamma_h.$$

Это следует из уравнений (27) и (28). Поскольку $\|\psi\|_\infty \leqslant c\,h^2$, где постоянная c не зависит от h, то, согласно с устойчивостью схемы,

$$\|u - y\|_\infty = \|z\|_\infty \leqslant L^2/2\,\|\psi\|_\infty \leqslant c\,L^2/2\,h^2 = C\,h^2. \quad \square$$

Задания для самостоятельной работы

ВОПРОСЫ ДЛЯ САМОКОНТРОЛЯ

1. Запишите уравнение теплопроводности. а) Укажите ограничения на коэффициенты уравнения. б) Какие дополнительные условия приводят к однозначно разрешимой задаче?

2. В чем основная идея решения задачи методом прямых?

3. Укажите способ построения полудискретной задачи методом прямых для задачи теплопроводности.

4. Дайте формулировку явной разностной схемы для задачи теплопроводности.

5. Укажите условия устойчивости явной разностной схемы.

6. Дайте оценку погрешности аппроксимации явной разностной схемы.

7. Приведите оценку точности явной разностной схемы.

8. Какое количество арифметических операций требуется для определения решения явной разностной схемы при заданном t?

9. Дайте формулировку неявной разностной схемы для задачи теплопроводности.

10. Укажите условия устойчивости неявной разностной схемы.

11. Дайте оценку погрешности аппроксимации неявной разностной схемы.

12. Приведите оценку точности неявной разностной схемы.

13. Какое количество арифметических операций требуется для определения решения неявной разностной схемы при заданном t?

14. Дайте формулировку однородной задачи Дирихле для уравнения Пуассона в квадратной области.

15. Как определяется равномерная сетка с шагом h в квадратной области?

16. Дайте формулировку разностной схемы для уравнения Пуассона.

17. Приведите матричную формулировку разностной схемы для уравнения Пуассона. Является ли матрица системы а) симметричной? б) положительно определенной?

18. Является ли разностная схема для уравнения Пуассона а) устойчивой? б) однозначно разрешимой?

ЗАДАЧИ И УПРАЖНЕНИЯ

1. Для задачи Дирихле для уравнения теплопроводности

$$\frac{\partial u}{\partial t} - \frac{\partial^2 u}{\partial x^2} = f(x,t), \quad 0 < x < 1,\ t \in (0,T],$$

$$u(x,0) = u_0(x), \quad 0 < x < 1, \quad u(0,t) = u(1,t) = 0, \quad t \in (0,T],$$

а) постройте схему метода прямых. Запишите ее в виде задачи Коши для системы ОДУ;

б) аппроксимируйте полученную задачу Коши методом трапеций и получите разностную схему (называемую схемой Кранка – Николсона). Оцените погрешность аппроксимации этой схемы.

с) убедитесь, что эта схема является полусуммой явной и неявной схемы и докажите ее устойчивость в норме

$$\|y\| = \max_{j=0:M} \|y^j\|_2,$$

где $\|y^j\|_2$ евклидова норма вектора y^j.

d) получите оценку точности схемы.

Указание. Запишите схему в следующем матричном виде

$$\frac{y^{j+1} - y^j}{\tau} + A\Big(\frac{y^{j+1} + y^j}{2}\Big) = \frac{f^{j+1} + f^j}{2}, \quad j = 0 : M-1.$$

Умножая это равенство на $y^{j+1} - y^j$ и суммируя по всем $j = 0 : m$, $m \leqslant M-1$, получите оценку устойчивости. Воспользуйтесь положительной определенностью матрицы A.

2. Для однородной задачи Дирихле в прямоугольной области $\Omega = (0,L) \times (0,L)$:

$$-\Delta u(x) + g(x)u(x) = f(x), \quad x \in \Omega, \quad u(x) = 0, \quad x \in \partial\Omega,$$

где $g(x) \geqslant \alpha > 0$ для $\forall\, x \in \overline{\Omega}$:

а) постройте разностную схему на равномерной сетке;

б) укажите вид и свойства матрицы этой схемы;

с) докажите однозначную разрешимость построенной разностной схемы.

д) оцените ее погрешность аппроксимации.

Указание. При построении схемы и записи ее в матричном виде действуйте как и в случае $g(x) = 0$, внося требуемые изменения.

Литература

[1] Воеводин В.В. Вычислительные основы линейной алгебры — М.: Наука, 1977.

[2] Парлетт Б. Симметричная проблема собственных значений — М.: Мир, 1983.

[3] Хорн Р., Джонсон Ч. Матричный анализ — М.: Мир, 1989.

[4] Бахвалов Н.С., Жидков Н.П., Кобельков Г.М. Численные методы. — М.: Наука, 1987.

[5] Самарский А.А., Гулин А.В. Численные методы. — М.: Наука, 1989.

[6] Бахвалов Н. С., Лапин А. В., Чижонков Е. В. Численные методы в задачах и упражнениях. Учебное пособие. Под ред. В. А. Садовничего. — М.: Высшая школа. 2000.

[7] Богачев К. Ю. Практикум на ЭВМ. Методы решения линейных систем и нахождения собственных значений. — М.: Изд-во ЦПИ при механико-математическом ф-те МГУ. 1998.

[8] Тыртышников Е. Е. Методы численного анализа. — М.: 2006.

[9] Срочко В. А. Численные методы. Курс лекций. Учебное по- собие для вузов. Санкт-Петербург: Изд-во ЛАНЬ. 2010.

[10] Глазырина Л.Л., Карчевский М.М. Введение в численные методы — Казань, КФУ, 2012

Р.З. Даутов, М.Р. Тимербаев

Численные методы.
Решение задач линейной алгебры и дифференциальных уравнений

Учебное пособие