

## Оглавление

ВВЕДЕНИЕ .....	5
1. СИСТЕМЫ ЛИНЕЙНЫХ УРАВНЕНИЙ И МЕТОДЫ ИХ РЕШЕНИЯ.....	6
1.1.    Метод Гаусса.....	11
1.2.    Метод простых итераций .....	15
1.3.    Сравнительная оценка прямых и итерационных методов ..	20
2. НАХОЖДЕНИЕ СОБСТВЕННЫХ ЗНАЧЕНИЙ И СОБСТВЕННЫХ ВЕКТОРОВ МАТРИЦ.....	22
2.1.    Определение наибольшего и наименьшего собственных значений итерационным методом .....	25
3. АППРОКСИМАЦИЯ И ИНТЕРПОЛЯЦИЯ ФУНКЦИЙ .....	30
3.1.    Постановка задачи и основные определения.....	30
3.2.    Интерполяция с помощью многочленов.....	34
3.3.    Интерполяционный многочлен Лагранжа .....	34
3.4.    Интерполяционный многочлен Ньютона .....	37
3.5.    Точность и сходимость многочленной интерполяции .....	39
3.6.    Использование локальных интерполяций .....	41
4. ЧИСЛЕННОЕ ДИФФЕРЕНЦИРОВАНИЕ.....	43
4.1.    Постановка задачи .....	43
4.2.    Использование ряда Тейлора .....	45
5. ЧИСЛЕННОЕ ИНТЕГРИРОВАНИЕ .....	48
5.1.    Постановка задачи .....	48
5.2.    Метод прямоугольников .....	51
5.3.    Метод трапеций.....	53
5.4.    Метод Симпсона .....	56
6. ЧИСЛЕННЫЕ МЕТОДЫ РЕШЕНИЯ НЕЛИНЕЙНЫХ УРАВНЕНИЙ И СИСТЕМ НЕЛИНЕЙНЫХ УРАВНЕНИЙ.....	59
6.1.    Нелинейные уравнения .....	59
6.1.1.    Постановка задачи.....	59
6.1.2.    Отделение корней нелинейного уравнения .....	61
6.1.3.    Метод половинного деления и метод хорд .....	62
6.1.4.    Метод простой итерации .....	66
6.1.5.    Метод Ньютона .....	69
6.1.6.    Метод секущих .....	71
6.1.7.    Сравнительная оценка методов.....	72
6.2.    СИСТЕМЫ НЕЛИНЕЙНЫХ УРАВНЕНИЙ.....	74
6.2.1.    Постановка задачи и ее особенности.....	74
6.2.2.    Метод Ньютона и его модификации .....	77

7. ОБЫКНОВЕННЫЕ ДИФФЕРЕНЦИАЛЬНЫЕ УРАВНЕНИЯ .....	83
7.1. Общие понятия.....	83
7.2. Метод конечных разностей .....	86
7.3. Решение задачи Коши.....	89
7.4. Метод Эйлера .....	91
7.5. Метод Эйлера с пересчетом .....	92
7.6. Метод Рунге–Кутты.....	94
7.7. Многошаговые методы.....	97
Лабораторная работа № 1_Решение систем линейных алгебраических уравнений точными методами .....	100
Лабораторная работа № 2_Решение систем линейных алгебраических уравнений численными методами.....	103
Лабораторная работа № 3_Вычисление собственных значений и векторов матриц.....	105
Лабораторная работа № 4 Интерполирование функций .....	108
Лабораторная работа № 5_Дифференцирование функций .....	110
Лабораторная работа № 6_Приближенные методы вычисления собственных интегралов .....	112
Лабораторная работа № 7_Аппроксимация функций методом наименьших квадратов.....	118
Лабораторная работа № 8_Приближенные методы решения нелинейных уравнений и систем нелинейных уравнений.....	121
Лабораторная работа № 9_Численные методы решения обыкновенных дифференциальных уравнений .....	125
Заключение.....	128
Библиографический список .....	128

## ВВЕДЕНИЕ

В практической деятельности часто приходится иметь дело с математическими задачами, точное решение которых является довольно сложным или неизвестно. В этих случаях выходом из создавшейся ситуации являются приближенные вычисления. По этой причине приближенные методы решений имеют большое значение и широкое применение.

В настоящее время эффективное использование ЭВМ для решения научных и инженерных задач практически невозможно без достаточных знаний вычислительной математики. Кроме того, следует отметить, что при приближенном решении задач возникает вопрос необходимости оценки погрешности полученного результата.

Для практического усвоения рассматриваемого в курсе лекций теоретического материала авторами настоящего издания составлены соответствующие лабораторные работы. К каждой лабораторной работе приводится необходимый теоретический материал, обращение к которому позволит успешно выполнить индивидуальное задание.





– наблюдается большой рост вычислительной сложности алгоритма, например, для вычисления одного определителя путем вычисления всех входящих в сумму членов требуется около  $n \cdot n!$  арифметических операций; уже при  $n > 10$  время его работы становится практически нереализуемым;

– увеличение числа операций приводит к быстрому росту погрешности округления и может привести к переполнению разрядной сетки компьютера.

Кроме того, на практике приходится решать системы достаточно высокого порядка (от нескольких сотен до сотен тысяч уравнений), что приводит к недопустимому росту погрешности, если не использовать специальных приемов в процессе решения.

Если ввести понятие обратной матрицы на основании соотношения

$$A A^{-1} = E, \quad (1.6)$$

где  $A$  – исходная матрица;  $A^{-1}$  – обратная к исходной;  $E$  – единичная матрица, можно легко получить решение линейной системы (1.2). Действительно, умножив обе части (1.2) слева на  $A^{-1}$ , получим

$$A^{-1} A \bar{x} = A^{-1} \bar{b} \Rightarrow E \bar{x} = A^{-1} \bar{b} \Rightarrow \bar{x} = A^{-1} \bar{b}.$$

Следовательно, умножив матрицу  $A^{-1}$  на вектор правых частей  $b$ , мы получаем решение системы (1.2). Вообще говоря, нахождение обратной матрицы – задача более сложная, чем решение системы (1.2). Находить ее имеет смысл в случае, когда она представляет самостоятельный интерес для решаемой задачи, либо при необходимости находить ряд решений системы (1.2) с различными правыми частями.

Методы решения систем линейных уравнений можно разделить на две группы – *прямые (или точные) и итерационные*. **Прямые методы** – это методы, которые при отсутствии погрешностей округления за конечное число арифметических и логических операций дают точный результат. Эти методы сравнительно просты. Их иногда называют точными, поскольку решение выражается в виде точных формул через коэффициенты системы и правые части. Однако точным решение может быть лишь теоретически, при выполнении вычислений с бесконечным числом разрядов. На практике при использовании ЭВМ вычисления проводятся с ограниченным числом разрядов и поэтому неизбежны погрешности в окончательных результатах решения задачи.

**Итерационные методы** заключаются в построении последовательности приближенных решений, предел которой (если он существует) является решением системы.

Выбор конкретного численного метода определяется специфическими особенностями решаемой задачи и располагаемой вычислительной техникой. В первую очередь здесь необходимо учитывать вид матрицы решаемой линейной системы. Приведем наиболее типичные виды матриц, которые встречаются в различных практических задачах и являются основой многих методов.

**Диагональная матрица** – квадратная матрица, у которой  $a_{ij} = 0$  при  $i \neq j$  ( $i = 1, 2, \dots, n; j = 1, 2, \dots, n$ ).

**Единичная матрица** – диагональная матрица, у которой на главной диагонали стоят единицы, т.е.  $a_{ii} = 1$  ( $i = 1, 2, \dots, n$ ),  $a_{ij} = 0$  при  $i \neq j$ .

**Верхняя треугольная матрица** – матрица, у которой ниже главной диагонали все элементы равны нулю, т.е.  $a_{ij} = 0$  при  $i > j$ .

**Нижняя треугольная матрица** – матрица, у которой выше главной диагонали все элементы равны нулю, т.е.  $a_{ij} = 0$  при  $i < j$ .

**Симметричная матрица** – квадратная матрица, у которой все элементы, симметричные относительно главной диагонали, равны, т.е.  $a_{ij} = a_{ji}$  ( $i = 1, 2, \dots, n; j = 1, 2, \dots, n$ ).

**Ленточная матрица** – квадратная матрица, у которой все ненулевые элементы образуют “ленту”, параллельную главной диагонали, а все остальные элементы матрицы равны нулю, т.е.  $a_{ij} = 0$  при  $|i - j| \geq l$ , где  $l$  – ширина ленты.

**Трехдиагональная матрица** – квадратная матрица, у которой не равны нулю только те элементы, которые расположены на главной диагонали и двух соседних с ней диагоналях, т.е.  $a_{ij} \neq 0$  при  $|i - j| \leq 1$ .

Трехдиагональная матрица – частный случай ленточной матрицы.

**Разреженная матрица** – квадратная матрица, у которых количество ненулевых элементов гораздо меньше общего количества элементов.

В табл. 1.1 приведены примеры описанных матриц.

Если рассмотреть матрицу как упорядоченный набор чисел, который нужно хранить в памяти компьютера, то их можно разделить на два класса – хранимые матрицы и воспроизводимые матрицы.

Матрица называется **хранимой**, если при проведении расчетов ее элементы необходимо хранить в памяти компьютера. Матрица называется **воспроизводимой** (вычисляемой), если ее элементы не требуется хранить в памяти, так как в процессе вычислений по мере необходимости элементы матрицы могут быть вычислены по простым формулам (воспроизведены). Примером воспроизводимой матрицы является следующая трехдиагональная матрица:

$$A = \begin{pmatrix} 5 & 1 & 0 & 0 & 0 \\ -1 & 5 & 1 & 0 & 0 \\ 0 & -1 & 5 & 1 & 0 \\ 0 & 0 & -1 & 5 & 1 \\ 0 & 0 & 0 & -1 & 5 \end{pmatrix}$$

Ее элементы легко вычисляются:  $a_{ii} = 4$  на главной диагонали, т.е. при  $i = j$ ;  $a_{ij} = 1$  при  $j = i + 1$  (выше главной диагонали);  $a_{ij} = -1$  при  $j = i - 1$  (ниже главной диагонали);  $a_{ij} = 0$  в остальных случаях.

Таблица 1.1

### Примеры различных матриц

Тип матрицы	Пример
Диагональная	$\begin{pmatrix} 5 & 0 & 0 & 0 \\ 0 & -7 & 0 & 0 \\ 0 & 0 & 8 & 0 \\ 0 & 0 & 0 & 4 \end{pmatrix}$
Единичная	$\begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}$
Верхняя треугольная матрица	$\begin{pmatrix} 5 & 1 & 0.8 & 5 \\ 0 & -7 & 1 & 2 \\ 0 & 0 & 8 & 0 \\ 0 & 0 & 0 & 4 \end{pmatrix}$
Нижняя треугольная матрица	$\begin{pmatrix} 5 & 0 & 0 & 0 \\ & -7 & 0 & 0 \\ 3 & 4 & 8 & 0 \\ 1 & 9 & -4 & 4 \end{pmatrix}$
Симметричная матрица	$\begin{pmatrix} 5 & 10 & 5 & 7 \\ 10 & - & 0 & 9 \\ 5 & 0 & 8 & 0 \\ 7 & 9 & 0 & 4 \end{pmatrix}$
Ленточная матрица	$\begin{pmatrix} 5 & 1 & 2 & 0 & 0 & 0 \\ 9 & -7 & 5 & 1 & & 0 \\ 6 & 7 & 8 & 1 & 2 & 0 \\ 0 & 4 & 5 & 4 & 3 & 0 \\ 0 & 0 & 4 & 5 & 6 & 1 \\ 0 & 0 & 0 & 5 & 8 & 4 \end{pmatrix}$





Последним будет найдено значение неизвестной  $x_1$  из первого уравнения системы (1.8).

Таким образом, метод Гаусса состоит из двух шагов: прямого и обратного хода. Прямой ход заключается в сведении системы (1.2) к треугольному виду (1.8). Обратный ход заключается в последовательном вычислении искомых переменных из системы вида (1.8).

**Пример 1.1.** Методом Гаусса решить систему уравнений

$$\begin{cases} 2x_1 + x_2 - x_3 = 3, \\ x_1 - x_2 + 2x_3 = 1, \\ 3x_1 + 2x_2 + x_3 = 8. \end{cases} \quad (1.9)$$

Для того чтобы исключить из второго и третьего уравнений переменную  $x_1$ , из второго уравнения почленно вычтем первое уравнение, умноженное на  $1/2$ , а из третьего уравнения вычтем первое, умноженное на  $3/2$ . Такие преобразования приводят к эквивалентной системе, имеющей то же решение, что и исходная система. Подчеркнем, что при умножении уравнения на число производится умножение как правой, так и левой части уравнения. Преобразованная система имеет вид

$$\begin{cases} 2x_1 + x_2 - x_3 = 3, \\ 0 - \frac{3}{2}x_2 + \frac{5}{2}x_3 = -\frac{1}{2}, \\ 0 + \frac{1}{2}x_2 + \frac{5}{2}x_3 = \frac{7}{2}. \end{cases} \quad (1.10)$$

Для того чтобы привести систему к треугольному виду, нам осталось исключить  $x_2$  из третьего уравнения (1.10). Для этого из третьего уравнения вычтем второе уравнение, умноженное на  $-1/3$ . В результате получим

$$\begin{cases} 2x_1 + x_2 - x_3 = 3, \\ 0 - \frac{3}{2}x_2 + \frac{5}{2}x_3 = -\frac{1}{2}, \\ 0 + 0 + \frac{10}{3}x_3 = \frac{10}{3}. \end{cases} \quad (1.11)$$

На этом заканчивается прямой ход метода Гаусса, так как матрица системы (1.11) является верхней треугольной. С помощью обратного хода находим решение: из третьего уравнения находим  $x_3 = 1$ . Подставив это значение во второе уравнение, найдем, что  $x_2 = 2$ . Подстав-



Описанный процесс продолжается до тех пор, пока исходная система (1.1) не примет треугольный вид (1.8).

При описании прямого хода метода Гаусса предполагалось, что на каждом шаге в качестве ведущих были приняты элементы, стоящие на главной диагонали и не равные нулю. Покажем, что это ограничение не существенно.

Действительно, если, например,  $a_{11} = 0$ , то мы выберем такое  $i$ -е уравнение, в котором коэффициент при  $x_1$  не равен нулю ( $a_{i1} \neq 0$ ). Такое уравнение обязательно найдется, так как в противном случае весь первый столбец матрицы системы (1.1) состоит из нулей и, следовательно, определитель этой системы равен нулю и матрица  $A$  является вырожденной. Поменяем теперь местами первое и  $i$ -е уравнения и, приняв за ведущий элемент  $a_{i1}$ , проделаем все описанные выше и исключим  $x_1$  из всех уравнений, кроме  $i$ -го (которое теперь является первым). Аналогично можно поступать и в случае, когда какой-либо ведущий элемент на другом шаге равен нулю.

Существуют различные видоизменения метода Гаусса, которые, сохраняя его основную идею, отличаются способом хранения матриц коэффициентов в памяти ЭВМ, способом исключения элементов, методом "борьбы" с погрешностями округления. Имеются различные модификации метода для решения задач с матрицами специального вида (симметричными, ленточными и т.д.).

Доказано, что метод Гаусса – один из наиболее экономичных прямых методов для систем общего вида с точки зрения количества требуемых операций. В общем случае для решения системы  $n$  уравнений требуется приблизительно  $n^3/3$  операций. Для понимания возможностей метода Гаусса оценим на примере влияние порядка системы  $n$  на общее время решения системы и величину необходимой оперативной памяти. Посчитаем, сколько времени понадобится для решения  $10^4$  уравнений с плотно заполненной матрицей на компьютере со средним быстродействием  $10^6$  операций в секунду. Требуемое количество арифметических операций приблизительно будет равно  $N \approx 1/3 \cdot 10^{12}$ , следовательно, время решения задачи на компьютере примерно равно  $t \approx 1/3 \cdot 10^{12} / 10^6 \text{ с} = 1/3 \cdot 10^6 \text{ с} \approx 93 \text{ ч} \approx 3,8 \text{ сут.}$

Для хранения матрицы потребуется примерно  $4 \cdot n^2$  байтов, что в рассматриваемом случае составляет  $4 \cdot 10^8$  байт = 400 гигабайт (предполагается, что для хранения одного числа используется 4 байта). Для сравнения укажем, что объем памяти винчестера компьютера Pentium2 может варьироваться от 2 до 10 гигабайт, следовательно, решение подобной задачи на таких компьютерах не представляется реальным. Для

случая линейной системы при  $n = 10^3$  мы получим время счета порядка  $t \approx \frac{1}{3} \cdot \frac{10^9}{10^6} \text{ с} \approx 0,1 \text{ ч} = 6 \text{ мин}$  и объем памяти  $4 \cdot 10^6 \text{ байт} = 4 \text{ гигабайта}$ .

Эти оценки показывают, что на современных персональных компьютерах методом Гаусса можно решать линейные системы с плотно заполненной матрицей порядка  $n \approx 1000$ . С другой стороны, во многих научных и инженерных задачах приходится решать системы порядка десятков и сотен тысяч уравнений. Это удастся делать потому, что эти матрицы в подавляющем большинстве случаев являются сильно разреженными и воспроизводимыми, что позволяет для их решения использовать либо специальные модификации метода Гаусса либо итерационные методы. Один из наиболее часто применяемых – метод Гаусса с частичным выбором ведущего элемента, идея которого состоит в том, что в качестве ведущего элемента в каждом рассматриваемом столбце выбирается максимальный по модулю элемент в этом столбце. Описанный выше алгоритм прямого хода метода Гаусса фактически реализуется следующим образом. На первом шаге в первом столбце отыскивается максимальный по модулю элемент. Пусть, например, это будет элемент  $a_{i1}$ . Поменяем  $i$ -е и первое уравнения местами, так что максимальный элемент окажется в первой строке и первом столбце. Затем алгоритм исключения применяется ко всем уравнениям, кроме первого (фактически это будет  $i$ -е уравнение). На следующем шаге мы должны исключить  $x_2$  во всех уравнениях, кроме первых двух. Рассмотрим во втором столбце все элементы, начиная со второго, и выберем из них наибольший по модулю элемент. Пусть это будет элемент  $a_{k2}$ . Поменяем местами второе и  $k$ -е уравнение, так что элемент  $a_{k2}$  станет теперь ведущим элементом, который расположен во второй строке и во втором столбце. Исключим  $x_2$  во всех уравнениях, кроме первых двух. Аналогичный процесс повторяется до тех пор, пока матрица системы не примет треугольный вид.

## 1.2. Метод простых итераций

В отличие от прямых итерационные методы решения СЛАУ дают возможность получить решение лишь приближенно на основе группы определенных, неоднократно повторяющихся операций. Однократное выполнение такой группы операций называется итерацией. Получение все более точного решения на основе многократного повторения итераций возможно в том случае, когда итерационный метод сходится к искомому решению. В методе итераций решение системы (1.1) строит-









$$\begin{pmatrix} 2 & 1 & -3 \\ 4 & -1 & -1 \\ 2 & 5 & 3 \end{pmatrix}$$

не выполняются условия диагонального преобладания  $|2| < |1| + |-3|$ ,  $|-1| < |4| + |-1|$ ,  $|3| < |2| + |5|$ . Однако можно заметить, что в каждом уравнении (1.21) есть такая переменная  $x_i$ , коэффициент при которой не меньше, чем сумма модулей коэффициентов при других неизвестных в этом уравнении. В первом уравнении – это переменная  $x_3$ , во втором –  $x_1$ , в третьем –  $x_2$ . Если теперь поменять местами уравнения (1.21), сделав второе уравнение первым, третье уравнение – вторым, а первое – третьим

$$\begin{cases} 4x_1 - x_2 - x_3 = 2, \\ 2x_1 + 5x_2 + 3x_3 = 10, \\ 2x_1 + x_2 - 3x_3 = 0. \end{cases} \quad (1.23)$$

то матрица этой системы

$$\begin{pmatrix} 4 & -1 & -1 \\ 2 & 5 & 3 \\ 2 & 1 & -3 \end{pmatrix} \quad (1.24)$$

будет иметь диагональное преобладание  $|4| > |-1| + |-1|$ ,  $|5| = |2| + |3|$ ,

$|-3| = |1| + |2|$ . Записав теперь систему (1.23) в виде удобном для проведения итераций

$$\begin{cases} x_1 = 1/4 \cdot (2 + x_2 + x_3), \\ x_2 = 1/5(10 - 2x_1 - 3x_3), \\ x_3 = 1/3 \cdot (0 + 2x_1 + x_2). \end{cases} \quad (1.25)$$

получим сходящийся итерационный процесс, численные результаты которого приведены в табл. 1.2.

Таблица 1.2

### Численные результаты итерационного процесса

$k$	0	1	2	3	4	5
$x_1^{(k)}$	0	0,5	1,183333	1,019444	0,978426	1,001651
$x_2^{(k)}$	0	1,8	0,966667	0,925556	1,015741	1,004821
$x_3^{(k)}$	0	0,933333	1,11111	0,988148	0,990864	1,002708

### 1.3. Сравнительная оценка прямых и итерационных методов

Выбор численного метода решения систем линейных уравнений зависит от вида решаемой задачи и типа имеющейся вычислительной техники. Для заполненных матриц сравнительно небольшой размерности ( $n < 1000$ ) одним из наиболее эффективных является, по-видимому, метод Гаусса и его различные модификации, так как он прост в реализации и наиболее универсален. Однако с ростом  $n$  начинают проявляться недостатки прямых методов, которые заключаются в следующем:

- они требуют, как правило, хранения всей матрицы, что является очень серьезным ограничением на объем требуемой оперативной памяти (можно, конечно, хранить всю матрицу на жестком диске, но тогда требуется организовать постоянный обмен массивов информации между оперативной памятью и диском, что значительно увеличивает общее время решения задачи);
- происходит постоянное накапливание погрешностей округления, так как на каждом этапе вычислений используются результаты предыдущих операций.

В случае, если матрица является разреженной, но ненулевые элементы располагаются в матрице упорядоченным образом (например, матрица является ленточной или ненулевые элементы имеют вид клеток, расположенных вдоль главной диагонали), то метод Гаусса также является достаточно эффективным, так как можно организовать программу реализации метода так, чтобы нулевые элементы не хранить в памяти ЭВМ.

Для разреженной матрицы произвольного вида метод Гаусса становится неудобным в силу того, что в процессе преобразований вместо некоторых нулевых элементов появляются ненулевые элементы, что приводит к значительному увеличению количества ненулевых элементов. В этом случае итерационные методы могут быть гораздо более эффективными, так как для них, как правило, не требуется хранить нулевые элементы и совершать с ними какие-либо операции. Более того, во многих случаях, встречающихся в практических задачах, можно вообще не хранить элементы матрицы в памяти ЭВМ, а вычислять их по мере необходимости. Другое достоинство итерационных методов заключается в том, что в них погрешность не накапливается в процессе вычислений, так как точность вычисления каждой итерации определяется лишь результатами предыдущей итерации и почти не зависит от результатов других итерационных шагов. Это свойство самокорректировки метода, когда допущенная на какой-либо итерации небольшая

ошибка не влияет на конечный результат, особенно полезно при решении систем линейных уравнений высокого порядка.

Недостатком итерационных методов во многих случаях является их довольно медленная сходимость, особенно при больших  $n$ . Поэтому в вычислительной математике большое внимание уделяется вопросам ускорения сходимости итерационных методов.

Во многих случаях оказывается удобным комбинировать применение прямых и итерационных методов. Одним из таких подходов является итерационное уточнение решения системы линейных уравнений, полученного каким-либо прямым методом. Изложим кратко суть этого метода.

Пусть имеется система линейных уравнений, записанная в матричном виде  $A\bar{x} = \bar{b}$  и  $x_*$  – приближенное решение, полученное каким-либо методом. Подставим  $x_*$  в уравнение и вычислим невязку уравнения  $r_* = b - Ax_*$ . Для точного решения  $x$  невязка равна нулю  $r = b - Ax$ . Для приближенного решения невязка будет достаточно малым числом. Построим погрешность решения  $\varepsilon_1 = x_* - x$  и получим уравнение для определения погрешности

$$A\varepsilon_1 = A(x_* - x) = Ax_* - Ax = b - (b - r) = r.$$

Приведенные выкладки показывают, что погрешность удовлетворяет исходному уравнению, в котором в правой части вместо вектора  $b$  стоит вектор невязки  $r$ , вычисленный подстановкой приближенного решения  $x$  в исходное уравнение. Определим из уравнения

$$A\varepsilon_1 = r \tag{1.26}$$

величину  $\varepsilon_1$ . Если бы величина  $\varepsilon_1$  определялась из (1.26) точно (без погрешностей округления), то мы определили бы точное решение  $x_* = x + \varepsilon_1$ . Однако из-за наличия погрешностей знание  $\varepsilon_1$  позволяет нам определить не точное решение, а уточненное значение решения

$$x_1 = x + \varepsilon_1. \tag{1.27}$$

Во многих случаях вычислительный алгоритм, основанный на соотношениях (1.26) и (1.27), позволяет существенно уточнить первоначальное приближенное решение  $x$  при условии, что невязка  $r$  вычисляется достаточно точно. Вообще говоря, процесс итерационного уточнения можно продолжать, определяя величины  $x_2, x_3, \dots$ , однако это требует все более точного вычисления невязок уравнения. Удобство использования метода итерационного уточнения заключается в том, что для определения  $\varepsilon_1$  приходится решать систему линейных уравнений с той же исходной матрицей  $A$ . Поэтому, если для решения исходной системы  $A\bar{x} = \bar{b}$  используется  $LU$ -разложение, то знание

треугольных матриц  $L$  и  $U$  позволяет без труда решить систему (1.26) с помощью небольшого количества операций ( $\sim n^2$  операций вместо примерно  $n^3$  операций в методе Гаусса).

## 2. НАХОЖДЕНИЕ СОБСТВЕННЫХ ЗНАЧЕНИЙ И СОБСТВЕННЫХ ВЕКТОРОВ МАТРИЦ

Нахождение собственных значений и собственных векторов матриц требуется во многих физических и технических задачах при исследовании устойчивости различных процессов, при определении устойчивости и колебаний различных инженерных сооружений. В вычислительной математике исследования по устойчивости алгоритмов и анализу сходимости различных вычислительных методов также требуют решения задач на собственные значения.

**Собственные значения** для матрицы  $n$ -го порядка – это  $n$  действительных или комплексных чисел, удовлетворяющих (2.1) и определяемых как корни характеристического многочлена данной матрицы и характеризующих свойства матрицы при различных вычислительных преобразованиях. Матрица  $A$   $n$ -го порядка имеет  $n$  собственных значений, которые мы обозначим через  $\lambda_1, \lambda_2, \dots, \lambda_n$ , и соответствующие им ненулевые собственные векторы  $\vec{u}_1, \vec{u}_2, \dots, \vec{u}_n$ . В общем случае собственные значения могут быть как действительными, так и комплексными числами. Если среди собственных значений встречаются два или несколько одинаковых, то они называются кратными. В этом случае собственные векторы, соответствующие кратным собственным значениям, объединяются в один вектор. По определению собственные вектора и собственные значения удовлетворяют соотношению

$$A\vec{u}_i = \lambda_i \vec{u}_i \quad (i = 1, 2, \dots, n), \quad (2.1)$$

где  $\vec{u}_i$  – ненулевые векторы. Смысл этого выражения заключается в том, что собственные векторы – это такие ненулевые векторы, которые матрица  $A$  преобразует в коллинеарные им векторы, при этом соответствующие им собственные значения являются коэффициентами растяжения (сжатия) длины исходного собственного вектора.

Обозначив координаты собственного вектора  $\vec{u}_i = \{u_{i1}, u_{i2}, \dots, u_{in}\}$  и на основании соотношения (2.1) можно утверждать, что нахождение собственного вектора  $\vec{u}_i$  сводится к решению системы линейных уравнений для нахождения его координат



векторов называется **частичной проблемой собственных значений**. Во многих практических задачах приходится решать **обобщенную проблему собственных значений**. Она определяется следующим образом. Пусть заданы невырожденные матрицы  $A$  и  $B$ . Требуется найти ненулевой вектор  $\vec{u}$  и число  $\lambda$ , удовлетворяющие равенству

$$A\vec{u} = \lambda B\vec{u}. \quad (2.7)$$

Тогда  $\vec{u}$  и  $\lambda$  соответственно называются обобщенным собственным вектором и обобщенным собственным значением.

Легко показать, что обобщенную задачу можно свести к обычной задаче на собственные значения и собственные векторы. Действительно, умножив обе части векторного равенства (4.15) слева на обратную матрицу  $B^{-1}$ , получим

$$B^{-1}A\vec{u} = \lambda B^{-1}B\vec{u} \Rightarrow B^{-1}A\vec{u} = \lambda\vec{u}.$$

Видно, что обобщенная задача сводится к обычной задаче для матрицы  $B^{-1}A$ .

Перейдем к вопросам численного нахождения собственных значений и собственных векторов. Эта задача является одной из наиболее сложных вычислительных задач алгебры. На первый взгляд может показаться, что определение собственных значений сводится к нахождению корней характеристического многочлена (2.4), однако сложность здесь заключается в том, что для произвольной матрицы (особенно в случае большой размерности) весьма затруднительно вычислить сами коэффициенты  $c_i$  характеристического многочлена. Дополнительные трудности возникают из-за того, что среди собственных значений часто встречаются кратные. Поэтому большинство численных методов основано не на получении характеристического многочлена (2.4) для исходной матрицы, а на различных преобразованиях этой матрицы с помощью соотношений типа (2.5) и ее сведению к более простому виду, откуда собственные значения определяются гораздо проще.

На практике для симметричных матриц чаще всего используются различные варианты метода вращений. Не приводя подробное описание метода, которое является достаточно громоздким, опишем лишь его суть. К исходной матрице  $A$  многократно применяется преобразование (2.5), в результате чего матрица сводится к гораздо более простому виду: к диагональной матрице или к трехдиагональной матрице. Для диагональной матрицы собственные значения могут быть вычислены сразу  $\lambda_1 = a'_{11}, \lambda_2 = a'_{22}, \dots, \lambda_n = a'_{nn}$ , а для трехдиагональной матрицы они вычисляются с помощью простых рекуррентных формул.

Матрица  $P$ , с помощью которой производится преобразование (2.5), подбирается таким образом, чтобы на каждом шаге преобразова-

ния в преобразованной матрице  $P^{-1}AP$  появлялись два нулевых элемента, симметричных относительно главной диагонали. Такой матрицей  $P$  является матрица вращения, имеющая вид

$$P = \begin{bmatrix} 1 & 0 & \dots & \dots & \dots & \dots & \dots & 0 \\ 0 & 1 & \dots & \dots & \dots & \dots & \dots & \dots \\ \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots \\ \dots & \dots & \dots & \cos \varphi & \dots & -\sin \varphi & \dots & \dots \\ \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots \\ \dots & \dots & \dots & -\sin \varphi & \dots & \cos \varphi & \dots & \dots \\ \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots \\ 0 & 0 & \dots & \dots & \dots & \dots & \dots & 1 \end{bmatrix} \quad (2.8)$$

Эта матрица отличается от единичной матрицы  $E$  только изменением четырех членов: два из них стоят на главной диагонали  $p_{ii} = p_{jj} = \cos \varphi$ , а два других симметричны относительно главной диагонали  $p_{ij} = p_{ji} = \sin \varphi$ .

Геометрический смысл умножения на матрицу (2.8) заключается в том, что при таком преобразовании осуществляется плоское вращение в  $n$ -мерном пространстве в плоскости, проходящей через оси  $i$  и  $j$  на угол  $\varphi$  (остальные оси координат при таком вращении остаются неподвижными). Угол поворота  $\varphi$  выбирается таким образом, чтобы два элемента в исходной матрице, симметричные относительно главной диагонали, обратились в нуль.

Существуют различные варианты метода вращений, которые отличаются тем, к какому виду сводится исходная матрица  $A$ , и способом организации вычислений.

### 2.1. Определение наибольшего и наименьшего собственных значений итерационным методом

В практических задачах часто нужны не все собственные значения, а лишь некоторые из них. Чаще всего (например, в вопросах устойчивости) требуется определить лишь минимальное (или максимальное) по модулю собственное значение матрицы. В этом случае нецелесообразно решать полную проблему собственных значений.

Для решения частичной проблемы собственных значений, состоящей в определении одного или нескольких собственных значений и соответствующих им собственных векторов, проще всего использовать итерационные методы.

Строится итерационный процесс, который сходится к одному собственному вектору с соответствующим вычислением одного собственного значения. Опишем такой процесс для нахождения максимального

по модулю собственного значения и соответствующего собственного вектора. Рассмотрим случай, когда матрица  $A$   $n$ -го порядка имеет все действительные собственные значения  $\lambda_1, \lambda_2, \dots, \lambda_n$  и соответствующие им собственные векторы  $\vec{u}_1, \vec{u}_2, \dots, \vec{u}_n$ . Пусть также все собственные значения расположены в порядке убывания, причем значение  $\lambda_1$  по модулю строго больше, чем остальные собственные значения

$$|\lambda_1| > |\lambda_2| \geq |\lambda_3| \geq \dots \geq |\lambda_n|. \quad (2.9)$$

Собственные значения матрицы обладают следующим свойством: если все они действительны и различны, то соответствующие им векторы образуют базис в  $n$ -мерном пространстве. Это значит, что любой  $n$ -мерный вектор можно представить в виде линейной комбинации векторов  $\vec{u}_1, \vec{u}_2, \dots, \vec{u}_n$ . Возьмем произвольный вектор  $\vec{y}_0$ . Тогда

$$\vec{y}_0 = c_1 \vec{u}_1 + c_2 \vec{u}_2 + \dots + c_n \vec{u}_n = \sum_{i=1}^n c_i \vec{u}_i. \quad (2.10)$$

Умножим теперь матрицу  $A$  на вектор  $\vec{y}_0$ . В результате вектор  $\vec{y}_0$  преобразуется в другой вектор  $\vec{y}_1$

$$\vec{y}_1 = A \vec{y}_0. \quad (2.11)$$

Запишем в развернутом виде вектор  $\vec{y}_1$ , используя (2.10), (2.11) и определение собственных векторов и собственных значений

$$A \vec{u}_i = \lambda \vec{u}_i (i = 1, 2, \dots, n). \quad (2.12)$$

Тогда (2.11) запишется в виде

$$\vec{y}_1 = A \vec{y}_0 = A \left( \sum_{i=1}^n c_i \vec{u}_i \right) = \sum_{i=1}^n c_i (A \vec{u}_i) = \sum_{i=1}^n c_i \lambda_i \vec{u}_i. \quad (2.13)$$

Формула (2.13) показывает, что действуя матрицей  $A$  на произвольный вектор  $\vec{y}_0$ , мы получаем вектор, коэффициенты разложения которого по собственным векторам умножаются на соответствующее данному собственному вектору собственное значение.

Умножим теперь матрицу  $A$  на полученный вектор  $\vec{y}_1$  и получим новый вектор  $\vec{y}_2$ :

$$\vec{y}_2 = A \vec{y}_1 = A^2 \vec{y}_0 = A \left( \sum_{i=1}^n c_i \lambda_i \vec{u}_i \right) = \sum_{i=1}^n c_i \lambda_i (A \vec{u}_i) = \sum_{i=1}^n c_i \lambda_i^2 \vec{u}_i.$$

Повторяя многократное умножение матрицы  $A$  на вновь получаемые векторы, получаем

$$\vec{y}_k = A \vec{y}_{k-1} = A^k \vec{y}_0 = \sum_{i=1}^n c_i \lambda_i^k \vec{u}_i. \quad (2.14)$$



Теперь воспользуемся тем, что  $\lambda_1$  максимальное по модулю собственное значение, и преобразуем (2.14)

$$\bar{y}_k = \lambda_1^k [c_1 \bar{u}_1 + c_2 \left(\frac{\lambda_2}{\lambda_1}\right)^k \bar{u}_2 + c_3 \left(\frac{\lambda_3}{\lambda_1}\right)^k \bar{u}_3 + \dots + c_n \left(\frac{\lambda_n}{\lambda_1}\right)^k \bar{u}_n].$$

Так как ввиду (2.9)

$$\left|\frac{\lambda_2}{\lambda_1}\right| < 1, \left|\frac{\lambda_3}{\lambda_1}\right| < 1, \dots, \left|\frac{\lambda_n}{\lambda_1}\right| < 1,$$

то с ростом степени  $k$  эти величины убывают и при  $k \rightarrow \infty$  все  $(\lambda_i/\lambda_1)^k \rightarrow 0$ . Это означает, что при достаточно больших  $k$  коэффициенты, на которые умножаются векторы  $\bar{u}_2, \bar{u}_3, \dots, \bar{u}_n$ , будут малы и, следовательно, вектор  $\bar{y}_k$  будет почти коллинеарен первому собственному вектору

$$\bar{y}_k = \lambda_1^k c_1 \bar{u}_1. \quad (2.15)$$

Приведенные соотношения (2.14), (2.15) являются основой степенного метода нахождения максимального собственного значения. **Степенной метод** – итерационный метод нахождения наибольшего по модулю собственного значения матрицы и соответствующего ему собственного вектора. Метод называется степенным, так как в нем вычисляются различные степени матрицы. Однако алгоритм его реализации на практике несколько отличается от того, который был описан выше. Это вызвано тем, что при достаточно большом  $\lambda_1$  длина получаемых по формуле (2.14) векторов сильно растет. Поэтому удобно реализовать степенной метод таким образом, чтобы на каждом шаге итерации вида (2.14) проводить нормировку получаемого вектора, делая его длину, равной единице (здесь мы пользуемся тем, что собственный вектор определяется с точностью до произвольного сомножителя и поэтому, разделив каждую компоненту вектора на его длину, мы получим вектор единичной длины). Один из удобных способов реализации степенного метода заключается в следующем.

Задается начальный вектор  $\bar{y}_0$  с координатами  $\{\bar{y}_{01}, \bar{y}_{02}, \dots, \bar{y}_{0n}\}$ . Вычисление следующего приближения – вектора  $\bar{y}_1$  – состоит из трех этапов.

1) вычисляется длина вектора  $\bar{y}_0$

$$d_0 = \sqrt{\sum_{i=1}^n (y_{0i})^2}; \quad (2.16)$$

2) производится нормировка вектора  $\bar{y}_0$  путем деления каждой его компоненты на величину  $d_0$

$$\vec{W}_0 = 1/d_0 \vec{y}_0 . \quad (2.17)$$

В результате вектор  $\vec{W}_0$  будет иметь длину, равную единице;

3) вычисляется вектор  $\vec{y}_1$  по формуле

$$\vec{y}_1 = A\vec{W}_0 . \quad (2.18)$$

На этом завершается одна итерация степенного метода.

В общем случае переход от  $\vec{y}_k$  к вектору  $\vec{y}_{k+1}$  совершается по формулам

$$d_k = \sqrt{\sum_{i=1}^n (y_{ki})^2} ; \quad (2.19)$$

$$\vec{W}_k = 1/d_k \vec{y}_k ; \quad (2.20)$$

$$\vec{y}_{k+1} = A\vec{W}_k . \quad (2.21)$$

Итерации проводятся до тех пор, пока не выполнится условие  $|d_{k+1} - d_k| < \varepsilon$ .

Тогда можно принять, что  $\vec{u}_1 \approx \vec{y}_k$ ,  $\lambda_1 \approx d_k$ . Это означает, что последовательность  $d_k$  сходится к максимальному собственному значению  $\lambda_1$ , а последовательность векторов  $\vec{y}_k$  — к собственному вектору  $\vec{u}_1$ , соответствующему значению  $\lambda_1$ .

**Пример 2.1.** Найти максимальное собственное значение и соответствующий собственный вектор для матрицы  $A$

$$A = \begin{pmatrix} 10 & 1 \\ 1 & 2 \end{pmatrix}.$$

*Решение.* Задаем начальный вектор  $\vec{y}_0 = \{1, 0\}$ .

*1-й шаг.* Вычисляем длину вектора

$\vec{y}_0$  —  $d_0 = \sqrt{1+0} = 1$ . Для вектора  $\vec{y}_0$  нормировка не требуется, так как он имеет единичную длину. Поэтому  $\vec{W}_0 = \vec{y}_0$ . Переходим к вычислению  $\vec{y}_1 = A\vec{W}_0$

$$\vec{y}_1 = \begin{pmatrix} 10 & 1 \\ 1 & 2 \end{pmatrix} \begin{pmatrix} 1 \\ 0 \end{pmatrix} = \begin{pmatrix} 10 \\ 1 \end{pmatrix}.$$

*2-й шаг.* Вычисляем длину вектора  $\vec{y}_1$ :  $d_1 = \sqrt{10^2 + 1} = 10,05$ . Нормируем вектор  $\vec{y}_1$ , вычисляя  $\vec{W}_1$ :

$$\vec{W}_1 = \frac{1}{d_1} \vec{y}_1 = \frac{1}{10,05} \begin{pmatrix} 10 \\ 1 \end{pmatrix} = \begin{pmatrix} 0,9950 \\ 0,0995 \end{pmatrix}.$$

Умножаем матрицу  $A$  на полученный вектор  $\vec{W}_1$ :

$$\vec{y}_2 = A\vec{W}_1 = \begin{pmatrix} 10 & 1 \\ 1 & 2 \end{pmatrix} \begin{pmatrix} 0,9950 \\ 0,0995 \end{pmatrix} = \begin{pmatrix} 10,05 \\ 1,194 \end{pmatrix}.$$

3-й шаг. Вычисляем длину вектора  $\vec{y}_2$ :

$d_2 = \sqrt{10,05^2 + 0,0995^2} = \sqrt{102,428} = 10,1207$ . Нормируем вектор  $\vec{y}_2$ , вы-

числяя  $\vec{W}_2$ : 
$$\vec{W}_2 = \frac{1}{10,1207} \begin{pmatrix} 10,05 \\ 1,194 \end{pmatrix} = \begin{pmatrix} 0,993 \\ 0,118 \end{pmatrix}.$$

Умножаем матрицу  $A$  на полученный вектор  $\vec{W}_2$ :

$$\vec{y}_3 = A\vec{W}_2 = \begin{pmatrix} 10 & 1 \\ 1 & 2 \end{pmatrix} \begin{pmatrix} 0,993 \\ 0,118 \end{pmatrix} = \begin{pmatrix} 10,05 \\ 1,229 \end{pmatrix}.$$

4-й шаг. Вычисляем длину вектора  $\vec{y}_3$  и нормируем его:

$$d_3 = \sqrt{10,05^2 + 1,229^2} = 10,125.$$

$$\vec{W}_3 = \frac{1}{10,125} \begin{pmatrix} 10,05 \\ 1,229 \end{pmatrix} = \begin{pmatrix} 0,9926 \\ 0,1214 \end{pmatrix};$$

$$\vec{y}_4 = A\vec{W}_3 = \begin{pmatrix} 10 & 1 \\ 1 & 2 \end{pmatrix} \begin{pmatrix} 0,9926 \\ 0,1214 \end{pmatrix} = \begin{pmatrix} 10,047 \\ 1,235 \end{pmatrix}.$$

Видно, что величины  $d_2$  и  $d_3$  отличаются мало ( $|d_2 - d_3| = 0,0043$ ), поэтому можно считать, что наибольшее собственное значение и соответствующий собственный вектор равны

$$\lambda_1 \approx d_3 = 10,125, \quad \vec{u}_1 \approx \vec{W}_3 = \begin{pmatrix} 0,9926 \\ 0,1214 \end{pmatrix}.$$

Для сравнения укажем, что максимальное собственное значение, полученное с помощью характеристического уравнения, равно  $\lambda_{\max} = 10,123$ , а  $\lambda_{\min} = 1,887$ .

Отметим, что скорость сходимости степенного метода зависит от того, насколько  $\lambda_{\max}$  превышает остальные собственные значения. В рассмотренном выше примере  $\lambda_{\max} / \lambda_{\min} \approx 5$  и поэтому степенной метод быстро сходится. Фактически уже второй шаг дает хорошее приближение для  $\lambda_1$  и  $\vec{u}_1$ .

Часто бывает нужно определить не наибольшее, а наименьшее собственное значение матрицы  $A$ . В этом случае удобно использовать со-

отношение (2.6), согласно которому собственные значения обратной матрицы  $A^{-1}$  обратно пропорциональны собственным значениям матрицы  $A$ .

### 3. АППРОКСИМАЦИЯ И ИНТЕРПОЛЯЦИЯ ФУНКЦИЙ

#### 3.1. Постановка задачи и основные определения

Цель рассматриваемых задач – найти аналитическое выражение (формулу) для описания функции  $y = f(x)$ . Далее эта формула может быть использована для исследования функции методами математического анализа (например, для вычисления производных, интегралов и т.д.).

Существуют два основных случая, когда на практике необходимо получение формулы для написания функции.

Первый и наиболее часто встречающийся случай заключается в том, что функция  $y = f(x)$  задана таблично. Это означает, что дискретному множеству значений аргумента  $X\{x_0, x_1, \dots, x_n\}$  поставлено в соответствие множество значений функции  $Y\{y_0, y_1, \dots, y_n\}$ . Численные значения  $x_i$  и  $y_i$  могут быть определены либо в результате эксперимента, либо путем численных расчетов. В этом случае аналитическое описание функции дает единственный способ вычисления функции  $y = f(x)$  при любых значениях  $x \neq x_i$  и для вычисления производных и интегралов от этой функции.

Второй случай заключается в том, что аналитическая зависимость  $y = f(x)$  известна, но имеет очень сложный вид (например, в описание  $f(x)$  входят очень громоздкие функции, вычисление которых на компьютере занимает много времени). В этом случае имеет смысл найти более простое аналитическое описание функции  $y = f(x)$ , которое позволит значительно уменьшить время ее вычисления. При этом, конечно, необходимо потребовать, чтобы полученная более простая функция была достаточно близкой к исходной. Понятие близости этих функций должно быть определено в зависимости от типа конкретной практической задачи.

**Задача о приближении (аппроксимации) функций может быть сформулирована следующим образом.** Пусть задана функция  $f(x)$  и требуется приближенно заменить (аппроксимировать) функцию  $f(x)$  с помощью другой функции  $\varphi(x)$  таким образом, чтобы отклонение  $\varphi(x)$  от  $f(x)$  (в некотором определенном смысле) было наименьшим в

заданной области. Обычно функцию  $f(x)$  называют аппроксимируемой, а функцию  $\varphi(x)$  – аппроксимирующей.

Аппроксимация может быть точечной или непрерывной. Аппроксимация называется **точечной**, если аппроксимирующая функция строится на дискретном множестве точек  $X\{x_0, x_1, \dots, x_n\}$ . Аппроксимация называется непрерывной (или интегральной), если  $\varphi(x)$  строится на непрерывном множестве точек (например, на отрезке  $[a, b]$ ).

При постановке задачи аппроксимации должны быть решены следующие вопросы:

1. *Как выбирать узловые точки  $x_i \in X$ , на базе которых строится аппроксимация, если имеется некоторая свобода в их выборе? В некоторых случаях удобно использовать равноотстоящие значения аргумента, т.е. полагать  $x_i - x_{i-1} = \text{const}$  для всех значений  $i$ . Но иногда за счет специального распределения узлов  $x_i$  аппроксимируемой функции  $f(x)$  можно уменьшить погрешность аппроксимации и даже добиться сходимости в том случае, когда она отсутствует при равномерном распределении узловых точек.*

2. *Какой класс функций выбрать при построении аппроксимирующей функции  $\varphi(x)$ ? На практике чаще всего в качестве аппроксимирующей функции используются многочлены вида*

$$\varphi(x) = a_0 + a_1x + a_2x^2 + \dots + a_mx^m. \quad (3.1)$$

Такие приближения называют **многочленными приближениями**.

Коэффициенты многочлена  $a_i$  и степень многочлена  $m$  подбираются таким образом, чтобы обеспечить наименьшее отклонение  $\varphi(x)$  от  $f(x)$ . Вопрос о том, что понимать под наименьшим отклонением, зависит от конкретного вида решаемой задачи и будет разобран ниже. Удобство использования многочленов является очевидным. Во-первых, значения многочленов легко вычислять (например, с помощью схемы Горнера). Во-вторых, многочлены легко складывать, умножать, дифференцировать и интегрировать.

Другой класс аппроксимирующих функций – это тригонометрические многочлены

$$\varphi(x) = a_0 / 2 + \sum_{k=1}^m (a_k \cos(kx) + b_k \sin(kx)). \quad (3.2)$$

Приближения вида (3.2) называют рядами Фурье, теория которых достаточно хорошо разработана и которые широко используются в практических задачах.

3. *Каким должен быть критерий близости аппроксимируемой функции  $f(x)$  и аппроксимирующей ее функции  $\varphi(x)$ ?* Выбор критерия определяется существом решаемой задачи и тем, насколько точно задана исходная информация о функции  $f(x)$  (например, тем, точно или приближенно известна таблица значений аргумента и функции  $f(x)$  при табличном задании этой функции) и как далее нужно использовать полученную аппроксимирующую функцию  $\varphi(x)$ . Наиболее часто используются следующие критерии близости функций  $f(x)$  и  $\varphi(x)$ :

а) *совпадение функций  $f(x)$  и  $\varphi(x)$  на заданном дискретном множестве точек  $x_i$  ( $i = 0, 1, \dots, n$ ), на котором задана исходная функция  $f(x)$*

$$\varphi(x_i) = f(x_i) \quad i = 0, 1, 2, \dots, n. \quad (3.3)$$

Такой тип точечной аппроксимации называется **интерполированием**. Обычно точки  $x_i$  называют **узлами интерполяции**, а функцию  $\varphi(x)$ , если она ищется в виде многочлена, – **интерполяционным многочленом**. Таким образом, близость интерполяционного многочлена  $\varphi(x)$  к исходной функции  $f(x)$  состоит в том, что их значения совпадают на заданном дискретном множестве точек  $x_i$ , но могут отличаться при  $x \neq x_i$ , где  $x \in [x_0, x_n]$ . Если интерполяционный многочлен ищется в виде (3.1) на всем интервале  $[x_0, x_n]$ , на котором задана функция  $f(x)$ , то  $m = n$  и коэффициенты многочлена  $a_0, a_1, \dots, a_n$  в формуле (3.1) ищутся из условий (3.3), которые в совокупности образуют систему линейных уравнений для величин  $a_i$ . Недостаток интерполяции проявляется по крайней мере в двух случаях: при большом количестве узлов интерполяции, когда степень многочлена  $\varphi(x)$  становится большой; в том случае, когда табличные значения функции  $f(x)$  получены экспериментальным путем или путем не очень точных вычислений и содержат в себе ошибки. В последнем случае не имеет смысла требовать обязательного прохождения аппроксимирующего многочлена  $\varphi(x)$  через узловые точки  $x_i$ ,  $y_i = f(x_i)$  и удобнее применять другие критерии близости  $f(x)$  и  $\varphi(x)$ , один из которых приведен ниже;

б) *критерий близости  $f(x)$  и  $\varphi(x)$ , определяемый среднеквадратичным приближением*, в котором мерой близости функций  $f(x)$  и  $\varphi(x)$  является величина  $S$ , определяемая формулой

$$S = \sum_{i=0}^n [\varphi(x_i) - y_i]^2. \quad (3.4)$$

Величина  $S$  представляет собой сумму квадратов отклонений аппроксимирующей функции  $\varphi(x)$  от исходной функции  $f(x)$  на заданной системе точек  $x_0, x_1, \dots, x_n$ . Для заданной функции  $f(x)$  естественно выбирать  $\varphi(x)$  так, чтобы величина  $S$ , определяемая по (3.4), была минимальна. В случае многочленного приближения, когда  $\varphi(x)$  берется в виде (3.1), коэффициенты многочлена  $a_0, a_1, \dots, a_m$  подбирают из условия, что величина  $S$  является минимальной. Такой подход определяет сущность аппроксимации на основе **метода наименьших квадратов**. Отметим, что в этом методе не требуется, чтобы функция  $\varphi(x)$  проходила через узлы, в которых задана исходная табличная функция  $f(x)$ , а близость функций  $\varphi(x)$  и  $f(x)$  понимается лишь “в среднем”, в смысле, даваемом формулой (3.4). При этом, конечно, в отдельных точках функции  $\varphi(x)$  и  $f(x)$  могут отличаться довольно значительно. Впрочем, такой же недостаток имеет и метод интерполяции, в котором в точках, расположенных между узлами интерполяции, функции  $\varphi(x)$  и  $f(x)$  могут значительно отличаться.

В некоторых практических задачах и теоретических исследованиях требуется, чтобы во всех точках отрезка  $[a, b]$ , на котором задана функция  $f(x)$ , отклонение многочлена  $\varphi(x)$  от функции  $f(x)$  было по абсолютной величине меньше, чем заданная малая величина  $\varepsilon > 0$ :

$$|\varphi(x) - f(x)| < \varepsilon \quad (x \in [a, b]). \quad (3.5)$$

В этом случае говорят, что  $\varphi(x)$  равномерно аппроксимирует функцию  $f(x)$  на отрезке  $[a, b]$ . Критерий близости функций  $\varphi(x)$  и  $f(x)$  здесь будет определяться понятием **абсолютного отклонения  $D$  функций  $\varphi(x)$  и  $f(x)$**

$$\Delta = \max |\varphi(x) - f(x)| \quad (x \in [a, b]). \quad (3.6)$$

Возможность построения многочлена, равномерно приближающего функцию  $f(x)$ , определяется теоремой Вейерштрасса об аппроксимации.

**Теорема.** Если функция  $f(x)$  непрерывна на отрезке  $[a, b]$ , то для любого  $\varepsilon > 0$  существует многочлен  $\varphi(x)$ , степени  $m$ , которого зависит от  $\varepsilon$ , т.е.  $m = m(\varepsilon)$ , абсолютное отклонение которого от функции  $f(x)$  на отрезке  $[a, b]$  меньше  $\varepsilon$ .

Различия трех рассмотренных критериев приближения функций – интерполяции, среднеквадратичного приближения и равномерного приближения – приведены на рис. 3.1.





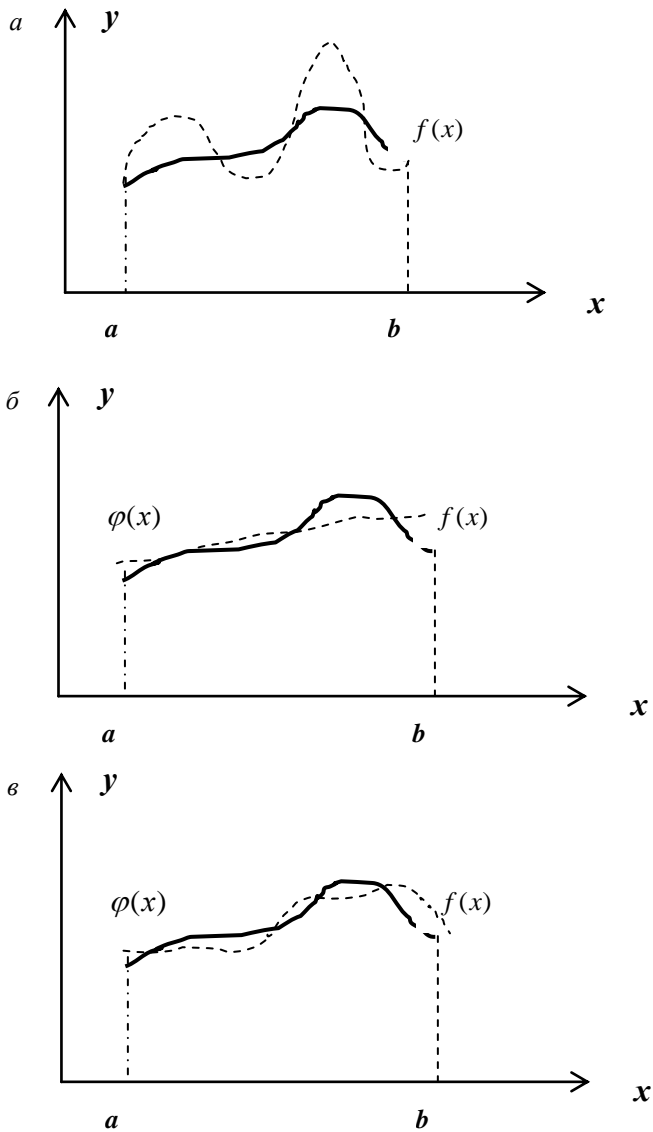


Рис.3.1. Критерии приближения функций.  
 $a$  – интерполяция;  $\bar{b}$  – среднеквадратичное приближение;  $\bar{c}$  – равномерное приближение

$L(x)$  представляет собой многочлен степени  $n$  от переменной  $x$ , в чем можно убедиться, перемножив двучлены, стоящие в числителе каждого члена суммы, и учитывая, что величины  $x_i$  и  $y_i$  являются заданными числами, определяющими таблично заданную функцию  $y = f(x)$ . Структура многочлена (3.8) такова, что он удовлетворяет основным условиям интерполяции: значения этого многочлена при  $x = x_i$  равны значениям интерполируемой функции  $y = f(x)$  в этих точках  $y_i = f(x_i)$ .

Можно показать, что для заданной функции  $f(x)$  этот многочлен является единственным. С помощью  $L(x)$  можно определить значения функции  $f(x)$  при любых значениях  $x$ , не являющихся узлами  $x_i$  функции  $f(x)$ . Из формулы (3.8), как частные случаи, можно получить формулы для **линейной и квадратичной интерполяции**, которые часто используются на практике.

**Линейная интерполяция** – простейший случай интерполяции многочленами, когда интерполирующей функцией является кусочно-линейная функция (ломаная).

При  $n = 1$

$$L(x) = \frac{x - x_1}{x_0 - x_1} y_0 + \frac{x - x_0}{x_1 - x_0} y_1. \quad (3.9)$$

Легко видеть, что  $L(x_0) = y_0$ ,  $L(x_1) = y_1$ . Выражение (3.9) дает формулу линейной интерполяции для приближенного определения значения  $f(x)$  при любом значении  $x$ , лежащем внутри интервала  $[x_0, x_1]$ , на концах которого значения функции известны:  $f(x_0) = y_0$ ,  $f(x_1) = y_1$ .

При  $n = 2$

$$L(x) = \frac{(x - x_1)(x - x_2)}{(x_0 - x_1)(x_0 - x_2)} y_0 + \frac{(x - x_0)(x - x_2)}{(x_1 - x_0)(x_1 - x_2)} y_1 + \frac{(x - x_0)(x - x_1)}{(x_2 - x_0)(x_2 - x_1)} y_2. \quad (3.10)$$

Видно, что  $L(x_0) = y_0$ ,  $L(x_1) = y_1$ ,  $L(x_2) = y_2$ . Квадратный многочлен (3.10) дает возможность по значениям  $y_0, y_1, y_2$  функции  $f(x)$  в трех заданных точках  $x_0, x_1, x_2$  определить приближенное значение  $f(x)$  в любой точке  $x \in [x_0, x_2]$ . Формула (3.10) называется формулой квадратичной интерполяции.

Приведем примеры использования формул (3.9) и (3.10).

**Пример 3.1.** Функция  $y = f(x)$  задана таблицей:

$x$	1	2	3
$y$	0	0,693147	1,0986123

Вычислить значение функции  $f(x)$  в точке  $x = 1,5$ , используя линейную и квадратичную интерполяцию.

*Решение.* В нашем случае  $x_0 = 1, x_1 = 2, x_2 = 3, y_0 = 0, y_1 = 0,69315, y_2 = 1,09861$ . Для вычисления  $f(1,5)$  воспользуемся формулой линейной интерполяции (3.9)

$$f(1,5) = \frac{1,5-1}{-1} \cdot 0 + \frac{1,5-1}{1} \cdot 0,69315 = 0,346574.$$

Воспользуемся теперь формулой квадратичной интерполяции (3.10)

$$\begin{aligned} f(1,5) &= \frac{(1,5-2)(1,5-3)}{(1-2)(1-3)} \cdot 0 + \frac{(1,5-1)(1,5-3)}{(2-2)(2-3)} \cdot 0,693147 + \\ &+ \frac{(1,5-1)(1,5-(-2))}{(3-1)(2-1)} \cdot 1,098612 = 0,382536. \end{aligned}$$

Для сравнения результатов укажем, что приведенная в таблице функция – это функция  $y = \ln x$ , в которой были использованы шесть знаков после запятой. Точное значение  $y(1,5)$  будет равно  $\ln(1,5) = 0,405465$ , линейная интерполяция дает  $y(1,5) = 0,346574$ , а квадратичная интерполяция –  $y(1,5) = 0,382536$ . Для линейной интерполяции ошибка равна  $0,058891$ , а для квадратичной –  $0,022929$ .

### 3.4. Интерполяционный многочлен Ньютона

В многочлене Лагранжа (3.8) используются произвольно расположенные узлы интерполяции  $x_i$ . Рассмотрим теперь часто встречающийся случай равноотстоящих узлов, когда  $x_i - x_{i-1} = h = \text{const}$  ( $i = 1, 2, \dots, n$ ). Величина  $h$  называется шагом. Получим для этого случая другой способ записи интерполяционного многочлена, называемый интерполяционным многочленом Ньютона. Подчеркнем здесь, что при заданном наборе узлов интерполяции интерполяционный многочлен является единственным. Поэтому различные виды интерполяционных многочленов (Лагранжа, Ньютона и многие другие) различаются лишь алгоритмами их получения и способом записи, представляя собой в сущности один и тот же многочлен. Удобство применения конкретного вида записи интерполяционного многочлена определяется существом решаемой задачи.

Для дальнейшего нам понадобится ввести понятие **конечных разностей**. В следующих разделах это понятие будет широко использоваться при изложении численных методов решения дифференциальных уравнений. Пусть известны значения функции  $y = f(x)$  в узлах  $x_i$ :  $y_i = f(x_i)$ . Составим разности значений функции в соседних узлах:

$$\begin{aligned}\Delta y_0 &= y_1 - y_0 = f(x_0 + h) - f(x_0), \\ \Delta y_1 &= y_2 - y_1 = f(x_0 + 2h) - f(x_0 + h),\end{aligned}\quad (3.11)$$

$$\dots\dots\dots$$

$$\Delta y_{n-1} = y_n - y_{n-1} = f(x_0 + nh) - f(x_0 + (n-1)h).$$

Величины  $\Delta y_i$  называются **первыми разностями функции**  $y = f(x)$ . Имея первые разности, можно составить **вторые разности функции**:

$$\begin{aligned}\Delta^2 y_0 &= \Delta y_1 - \Delta y_0 = y_2 - 2y_1 + y_0, \\ \Delta^2 y_1 &= \Delta y_2 - \Delta y_1 = y_3 - 2y_2 + y_1, \\ \dots\dots\dots \\ \Delta^2 y_i &= \Delta y_{i+1} - \Delta y_i = y_{i+2} - 2y_{i+1} + y_i.\end{aligned}\quad (3.12)$$

Аналогично вводятся разности  $k$ -го порядка

$$\Delta^k y_i = \Delta^{k-1} y_{i+1} - \Delta^{k-1} y_i. \quad (3.13)$$

Смысл введения разностей функции заключается в том, что они характеризуют скорость изменения функции и ее производных. Например, малые значения первых разностей означают, что функция  $y = f(x)$  в соответствующих точках изменяется незначительно, а большие значения означают, что функция сильно изменяется при переходе от  $x_i$  к  $x_{i+1}$ . Используя конечные разности, можно записать **интерполяционный многочлен Ньютона** в следующем виде:

$$\begin{aligned}N(x) &= y_0 + \frac{\Delta y_0}{h}(x - x_0) + \frac{\Delta^2 y_0}{2!h^2}(x - x_0)(x - x_1) + \dots + \\ &+ \frac{\Delta^n y_0}{n!h^n}(x - x_0)(x - x_1)\dots(x - x_{n-1}).\end{aligned}\quad (3.14)$$

Непосредственно по этой формуле вычисления проводить не очень удобно, так как и в числителе и в знаменателе стоят произведения малых величин. Обычно эту формулу записывают в другом виде, вводя новую переменную  $t = (x - x_0)/h$ . Тогда величины, входящие в (3.14), примут вид

$$\frac{x - x_1}{h} = t - 1, \quad \frac{x - x_2}{h} = t - 2, \dots, \quad \frac{x - x_{n-1}}{h} = t - n + 1. \quad (3.15)$$

Интерполяционный многочлен (3.14) теперь примет вид

$$N(x_0 + th) = y_0 + t\Delta y_0 + \frac{t(t-1)}{2!}\Delta^2 y_0 + \dots + \frac{t(t-1)\dots(t-n+1)}{n!}\Delta^n y_0. \quad (3.16)$$

Многочлен (3.16) позволяет определить значения функции  $y = f(x)$  при любом значении  $x \neq x_i$  на всем интервале  $[x_0, x_n]$ . Для этого

по величине  $x$  достаточно определить значение  $t = (x - x_0)/h$  и воспользоваться формулой (3.16), определив предварительно конечные разности функции  $y = f(x)$  по формулам (3.11) – (3.13). Однако чем меньше значение  $x$  при  $x \in [x_0, x_n]$ , тем меньше членов в сумме (3.16) приходится брать (значения слагаемых быстро убывают) и тем точнее получается результат интерполяции. Поэтому (3.16) целесообразно использовать при  $x_1 < x < x_2$  (таким образом,  $t < 1$ ). Выражение вида

$$N(x_i + th) = y_i + ty_i + \frac{t(t-1)}{2!} \Delta^2 y_i + \dots + \frac{t(t-1)\dots(t-n+1)}{n!} \Delta^n y_i \quad (3.17)$$

называется интерполяционным многочленом Ньютона для интерполирования вперед, его удобство использования заключается в том, что если взять в качестве базисной точки  $x_i$  левый узел интерполяции ( $x \in [x_i, x_{i+1}]$ ), можно получить ряд (3.17), в котором первое слагаемое будет главным членом, а остальные слагаемые будут носить характер поправок, причем значения поправок будут быстро уменьшаться для более далеких членов ряда.

В частном случае из формул (3.15) – (3.17) можно получить формулы для линейной или квадратичной интерполяции в случае равноотстоящих узлов. Из (3.15) при  $n = 1$ , когда  $x \in [x_0, x_1]$  получим формулу линейной интерполяции

$$y \approx y_0 + \frac{y_1 - y_0}{h}(x - x_0), \quad (3.18)$$

а при  $n = 2$ , когда  $x \in [x_0, x_2]$ , получим формулу квадратичной интерполяции

$$y \approx y_0 + \frac{y_1 - y_0}{h}(x - x_0) + \frac{y_2 - 2y_1 + y_0}{2h^2}(x - x_0)(x - x_1). \quad (3.19)$$

### 3.5. Точность и сходимость многочленной интерполяции

По определению интерполяции интерполяционные многочлены  $L(x)$  и  $N(x)$  совпадают в узлах интерполяции  $x_i$  со значениями функции  $y = f(x)$ . Однако при других  $x_i$  значения этих многочленов и функции  $f(x)$  не совпадают. Без вывода приведем формулы для оценки погрешности аппроксимации интерполяционных многочленов Лагранжа и Ньютона. Обозначим через  $L_n(x)$  и  $N_n(x)$  многочлены Лагранжа и Ньютона  $n$ -й степени. Рассмотрим погрешности интерполяции:

$$R_n(x) = f(x) - L_n(x); \quad (3.20)$$

$$Z_n(x) = f(x) - N_n(x). \quad (3.21)$$

Величины  $L_n(x)$  и  $N_n(x)$  называют остаточными членами многочленов Лагранжа и Ньютона. Можно показать, что для этих величин существуют следующие соотношения:

$$|R_n(x)| \leq \frac{(x-x_0)(x-x_1)\dots(x-x_n)}{(n+1)!} M_{n+1}, \quad (3.22)$$

где  $M_{n+1} = \max |f^{(n+1)}(x)|$  – максимальное значение производной порядка  $(n+1)$  функции  $y = f(x)$  на отрезке  $[x_0, x_n]$ ;

$$Z_n(x) = \frac{t(t-1)\dots(t-n)}{(n+1)!} f^{(n+1)}(\xi) h^{n+1}, \quad (3.23)$$

где  $\xi \in [x_0, x_n]$ ,  $t = (x-x_0)/h$ .

Оценка погрешности интерполяционных многочленов показывает, что с ростом числа узлов  $n$  и соответственно степени получаемого многочлена, погрешность может увеличиваться из-за роста  $|f^{(n+1)}(x)|$ . Кроме того, с ростом  $n$  увеличивается время вычисления многочлена на ЭВМ, и возрастает вычислительная погрешность. Это является одним из главных недостатков глобальной интерполяции, когда ищется единый интерполяционный многочлен на всем интервале  $[x_0, x_n]$ , где задана функция  $f(x)$ . Возможны два способа преодоления этих трудностей.

Первый способ основан на использовании локальной интерполяции, когда на каждом интервале  $[x_i, x_{i+1}]$  строится свой многочлен невысокой степени (обычно  $m = 1, 2$  или  $3$ ).

Второй способ основан на специальном расположении узлов интерполяции  $x_i$  (если они не фиксированы). Было установлено, что многочленная аппроксимация с равномерно расположенными узлами интерполяции  $x_i$  может приводить к совершенно неверным результатам при увеличении степени  $n$  интерполяционного многочлена. Впервые такой пример был приведен О. Рунге, который интерполировал функцию  $y = \frac{1}{1+25x^2}$  на интервале  $[-1, 1]$  с равномерным распределением узлов интерполяции. Оказалось, что при любых  $x$ , для которых  $|x| < 0,726$ , интерполяционный многочлен быстро сходится к соответствующему значению  $f(x)$  при  $n \rightarrow \infty$ , а на концах интервала при  $|x| > 0,726$  интерполяционный многочлен не стремится ни к какому

пределу и возрастает бесконечно в каждой точке этой части интервала при  $n \rightarrow \infty$ . Такое поведение интерполяционного многочлена при равномерном распределении узлов называется явлением Рунге. Таких затруднений можно избежать, если специальным образом выбирать неравномерно расположенные узлы интерполяции. В частности, показано, что если функция  $f(x)$  имеет непрерывную производную на отрезке  $[-1, 1]$ , то при выборе узлов интерполяции  $x_i$ , совпадающих с корнями многочленов Чебышева степени  $n+1$ , интерполяционные многочлены степени  $n$  сходятся к значениям  $f(x)$  в любой точке этого отрезка.

### 3.6. Использование локальных интерполяций

В силу отмеченных выше недостатков глобальной интерполяции одним многочленом на всем отрезке  $[x_0, x_n]$ , на практике часто используется локальная интерполяция, которую можно назвать кусочно-многочленной. Идея ее заключается в том, чтобы на каждом отдельном отрезке  $[x_i, x_{i+1}]$  использовать свой многочлен степени  $m$  ( $m < n$ ), коэффициенты которого определяются из условий совпадения значений этого многочлена в узловых точках со значениями функции  $f(x)$  в тех же точках и из некоторых условий сопряжения локальных многочленов в узловых точках с тем, чтобы обеспечить необходимую гладкость интерполирующей функции. Простейшим случаем локальной интерполяции является кусочно-линейная интерполяция, когда между узлами интерполяции функция ведет себя линейно (рис. 3.2). В этом случае на каждом интервале  $[x_i, x_{i+1}]$  используется уже рассмотренная ранее формула линейной интерполяции (3.9).

Наиболее часто используются кубические сплайны, в которых на каждом интервале  $[x_i, x_{i+1}]$  для интерполирования используется свой кубический многочлен, причем многочлены на соседних интервалах стыкуются определенным образом в узлах интерполяции  $x_i$ . Под термином сплайн подразумевается следующее понятие. Сплайном порядка  $m$  называют функцию  $S_m(x)$ , являющуюся многочленом порядка  $m$  на каждом из отрезков  $[x_i, x_{i+1}]$

$$S_m(x) = P_m(x) = a_{i0} + a_{i1}x + \dots + a_{im}x^m, x \in [x_i, x_{i+1}] \quad (3.24)$$

и удовлетворяющую условиям непрерывности производных до порядка  $m-1$  во внутренних узлах интерполяции  $x_1, x_2, \dots, x_{n-1}$ :

$$P_{m_i}^{(k)}(x_i) = P_{m_{i+1}}^{(k)}(x_i) \quad (3.25)$$

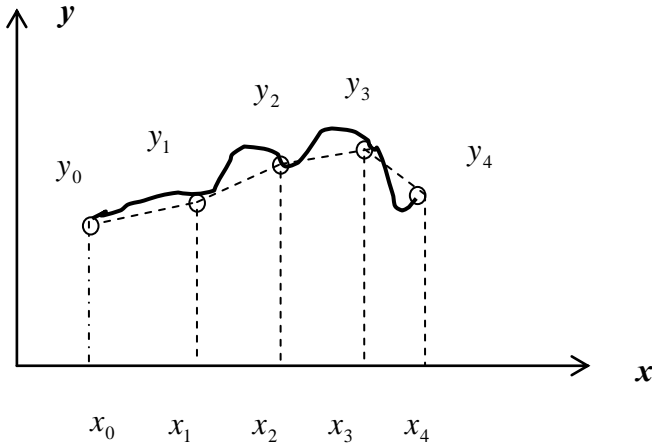


Рис.3.2. Геометрическая интерпретация линейной интерполяции

при  $k = 0, 1, 2, \dots, m-1$  ( $i = 1, 2, \dots, n-1$ ), где верхний индекс  $k$  означает порядок производной.

Кубический сплайн  $S_3(x)$  имеет следующий вид:

$$S_3(x) = P_{3i}(x) = a_{i0} + a_{i1}x + a_{i2}x^2 + a_{i3}x^3, \quad (3.26)$$

где коэффициенты кубического многочлена будут различными для каждого интервала  $[x_i, x_{i+1}]$ . Так как на общем интервале  $[x_0, x_n]$  всего  $n$  интервалов  $[x_i, x_{i+1}]$ , то общее количество неизвестных коэффициентов у многочленов равно  $4n$ . Для определения  $4n$  неизвестных могут быть написаны  $2n$  уравнений, выражающих условия прохождения функции  $S_3(x)$  через заданные точки

$$S_3(x_i) = y_i, S_3(x_{i+1}) = y_{i+1} \quad (3.27)$$

и  $2n - 2$  уравнения вида (3.25), выражающих условия непрерывности первых и вторых производных в узлах интерполяции (условия гладкого сопряжения многочленов на соседних интервалах). Два недостающих соотношения обычно задаются на концах интервала при  $x = x_0$  и  $x = x_n$  и выражают условия закрепления концов сплайна. В частности, при свободном закреплении концов полагают  $S_3''(x_0) = S_3''(x_n) = 0$ . Получаемая функция, называемая свободным куби-



ческим сплайном, является самой гладкой среди всех интерполяционных функций данного класса.

## 4. ЧИСЛЕННОЕ ДИФФЕРЕНЦИРОВАНИЕ

### 4.1. Постановка задачи

Под термином **численное дифференцирование** понимают приближенное вычисление производных для таблично заданной функции. Помимо этого основного случая, численное дифференцирование используется для вычисления производной аналитически заданной функции, когда производная имеет слишком сложное для вычислений выражение.

По определению производной функции  $y = f(x)$  называется предел отношения приращения функции к приращению аргумента, если последнее стремится к нулю

$$\frac{dy}{dx} = f'(x) = \lim_{\Delta x \rightarrow 0} \frac{\Delta y}{\Delta x}. \quad (4.1)$$

При численном дифференцировании можно воспользоваться формулой (4.1) без предельного перехода  $\Delta x \rightarrow 0$ , используя приближенное равенство

$$y' \approx \Delta y / \Delta x, \quad (4.2)$$

где  $\Delta x$  – конечное число, а  $\Delta y$  – конечная разность, вычисляемая по определенным правилам. В зависимости от способа вычисления  $\Delta y$  можно получить различные формулы получения производной в одной и той же точке. Пусть функция  $y = f(x)$  задана таблично: множеству значений аргумента  $x_0, x_1, \dots, x_n$  соответствует множество значений функции  $y_0, y_1, \dots, y_n$ . Введем конечные разности в точке  $x_1$ :  $\Delta y_1 = y_2 - y_1$  (правая разность);  $\Delta y_1 = y_1 - y_0$  (левая разность);  $\Delta y_1 = y_2 - y_0$  (центральная разность). Шаг изменения аргумента  $h = x_{i+1} - x_i = \text{const}$ . Тогда производную функции в точке  $x = x_1$  можно определить с помощью приближенных соотношений:

$$y'(x_1) = y'_1 \approx (y_2 - y_1)/h; \quad (4.3)$$

$$y'(x_1) = y'_1 \approx (y_1 - y_0)/h; \quad (4.4)$$

$$y'(x_1) = y'_1 \approx (y_2 - y_0)/2h. \quad (4.5)$$

Аналогично можно вычислить вторую производную

$$y''(x_1) = (y_1')' \approx \frac{y_2' - y_1'}{h} = \frac{(y_2 - y_1)/h - (y_1 - y_0)/h}{h} = \frac{y_2 - 2y_1 + y_0}{h}. \quad (4.6)$$

Такой подход к решению задачи численного дифференцирования является недостаточно общим, так как не позволяет получать многие необходимые для практики выражения для производных и не дает ответа на вопрос о погрешности производимых вычислений.

Для формулировки более общего подхода к задаче численного дифференцирования рассмотрим аппроксимацию данной функции  $f(x)$  некоторой аппроксимирующей функцией  $\varphi(x)$

$$f(x) = \varphi(x) + R(x). \quad (4.7)$$

В качестве аппроксимирующей функции  $\varphi(x)$  можно взять любую интерполяционную функцию. Функция  $R(x)$  в (4.7) – погрешность аппроксимации. Дифференцируя выражение (4.7), найдем значение производной

$$f'(x) = \varphi'(x) + R'(x). \quad (4.8)$$

В качестве приближенного значения производной  $f'(x)$  можно принять соответствующее значение производной  $\varphi'(x)$  интерполирующей функции:

$$f'(x) \approx \varphi'(x).$$

Функция  $R'(x) = f'(x) - \varphi'(x)$ , характеризующая величину отклонения приближенного значения производной от ее истинного значения, называется **погрешностью аппроксимации производной**.

Аналогично можно определить любую производную более высокого порядка

$$f^{(k)}(x) = \varphi^{(k)}(x) + R^{(k)}(x), \quad f^{(k)}(x) \approx \varphi^{(k)}(x). \quad (4.9)$$

Отметим принципиальную особенность задачи дифференцирования: она является некорректной в том смысле, что малые изменения значений функции могут привести к довольно большим различиям в значениях производных. Например, если рассмотреть две непрерывно дифференцируемые функции на отрезке  $[a, b]$ , связанные соотношением

$$\tilde{f}(x) = f(x) + \frac{1}{n} \cos(n^2 x). \quad (4.10)$$

Разность значений этих функций ограничена и при больших  $n$  может быть сделана сколь угодно малой. Действительно,  $|\tilde{f}(x) - f(x)| = \left| \frac{1}{n} \cos(n^2 x) \right| \leq \frac{1}{n} \rightarrow 0 (n \rightarrow \infty)$ . Это неравенство следует из

ограниченности функции косинуса при любом  $x$ :  $|\cos(n^2 x)| \leq 1$ . Возьмем теперь производную от обеих частей равенства (4.10)

$$\tilde{f}'(x) = f'(x) - \frac{n^2}{n} \sin(n^2 x) = f'(x) - n \sin(n^2 x),$$

откуда следует, что

$$|\tilde{f}(x) - f(x)| = n |\sin(n^2 x)|. \quad (4.11)$$

Из (4.11) следует, что для некоторых значений  $x$  разность производных  $f'(x)$  и  $\tilde{f}'(x)$  может оказаться равной  $n$  и, следовательно, может быть сколь угодно большой. Это фактически означает, что если мы аппроксимируем функцию  $f(x)$  с помощью аппроксимирующей функции  $\varphi(x)$  и имеют место равенства (4.7) и (4.8), то из малости погрешности аппроксимации не всегда следует малость погрешности аппроксимации производной. Этот факт необходимо учитывать при практическом применении численного дифференцирования, особенно в случае, когда исходная функция  $f(x)$  определяется экспериментально и, следовательно, величины  $y_i$  заданы с некоторой ошибкой. Для того чтобы эти ошибки не привели к значительным погрешностям в определении производных, применяются специальные приемы сглаживания исходной экспериментальной функции.

## 4.2. Использование ряда Тейлора

Пусть функция  $y = f(x)$  задана с помощью таблицы на отрезке  $[a, b]$ . Будем считать шаг изменения аргумента постоянным. Рассмотрим произвольную внутреннюю точку  $x_i$  и две соседних с ней точки  $x_{i-1}$  и  $x_{i+1}$ . Значения функции в этих точках обозначим через  $y_i, y_{i-1}, y_{i+1}$ . Будем считать функцию достаточно гладкой и разложим ее в ряд Тейлора в окрестности базовой точки  $x_i$ . С помощью этого ряда значения функции  $y = f(x)$  в соседних справа и слева точках  $x_{i-1}$  и  $x_{i+1}$  можно записать в следующем виде:

$$y_{i+1} = y_i + y_i' h + \frac{y_i'' h^2}{2!} + \frac{y_i''' h^3}{3!} + \frac{y_i^{(4)} h^4}{4!} + O(h^5); \quad (4.12)$$

$$y_{i-1} = y_i - y_i' h + \frac{y_i'' h^2}{2!} - \frac{y_i''' h^3}{3!} + \frac{y_i^{(4)} h^4}{4!} - O(h^5). \quad (4.13)$$

Здесь использованы обозначения:  $y_i' = y'(x_i)$ ;  $y_i'' = y''(x_i)$ ;  $y_i''' = y'''(x_i)$ . Переносим первый член суммы в правой части (4.12) в левую часть и разделив обе части на  $h$ , получим:

$$\frac{y_{i+1} - y_i}{h} = y'_i + \frac{y''_i}{2!}h + O(h^2) = y'_i + O(h). \quad (4.14)$$

Аналогично из (4.13) следует:

$$\frac{y_i - y_{i-1}}{h} = y'_i - \frac{y''_i}{2!}h + O(h^2) = y'_i - O(h). \quad (4.15)$$

Считая  $h$  малым и пренебрегая членами  $O(h)$ , получим конечно-разностные формулы для приближенной аппроксимации первой производной с помощью правой и левой разностей:

$$y'_i = \frac{y_{i+1} - y_i}{h}; \quad (4.16)$$

$$y'_i = \frac{y_i - y_{i-1}}{h}. \quad (4.17)$$

Вычитая почленно из (4.12) члены равенства (4.13) и разделив обе части полученного равенства на  $2h$ , получим

$$\frac{y_{i+1} - y_{i-1}}{2h} = y'_i + \frac{y''_i}{3!}h^2 + O(h^4) = y'_i + O(h^2). \quad (4.18)$$

Пренебрегая членом второго порядка малости  $O(h^2)$ , получим более точную аппроксимацию для первой производной, использующую центральную разность

$$y'_i = \frac{y_{i+1} - y_{i-1}}{2h}. \quad (4.19)$$

Соотношения (4.14), (4.15) и (4.18) позволяют получить выражения для погрешности аппроксимации первой производной по формулам (4.16), (4.17) и (4.19). Эти погрешности, соответственно, равны:

$$R = \frac{y''_i}{2}h, R = -\frac{y''_i}{2}h, R = \frac{y''_i}{6}h^2. \text{ Видно, что погрешность зависит от значения}$$

некоторой производной в базовой точке  $x_i$  и шага таблицы  $h$ . Зависимость от шага обычно записывают в виде  $O(h^k)$ . Показатель степени  $k$  называют **порядком погрешности аппроксимации производной** (или просто порядком аппроксимации). Равенства (4.14) и (4.15) показывают, что правые и левые разности дают первый порядок аппроксимации ( $k = 1$ ), а центральная разность дает второй порядок ( $k = 2$ ), что обеспечивает более высокую точность.

Получим теперь выражение для аппроксимации второй производной  $y''_i$ : сложив левые и правые части выражений (4.12) и (4.13), получим

$$y_{i+1} - y_{i-1} = 2y_i + y''_i h^2 + \frac{y^{(4)}_i}{12}h^4 + O(h^6).$$

Переносим  $2y_i$  в левую часть, разделив обе части на  $h^2$  и отбросив  $O(h^6)$ , получим

$$y_i'' = \frac{y_{i+1} - 2y_i + y_{i-1}}{h^2} - \frac{y_i^{(4)}h^2}{12}. \quad (4.20)$$

В формуле (4.20) в явном виде выписано выражение для погрешности аппроксимации (последний член в правой части). Видно, что эта аппроксимация имеет второй порядок точности по  $h$  и зависит от четвертой производной функции  $y = f(x)$  в базовой точке  $x_i$ . Считая эту погрешность малой, получим

$$y_i'' = \frac{y_{i+1} - 2y_i + y_{i-1}}{h^2}. \quad (4.21)$$

Приведем пример использования полученных формул.

**Пример 4.1.** Задана табличная функция  $y = f(x)$

$x$	0,2	0,3	0,4
$y$	0,008	0,027	0,064

Вычислить значения  $y_i'(x)$  при всех трех значениях аргументов и значение  $y_i''(x)$  в точке  $x = 0,3$ .

*Решение.* В данном случае  $h = 0,1$ . Для вычисления  $y'(0,2)$  используем формулу (4.16):

$$y'(0,2) = (0,027 - 0,008) / 0,1 = 0,019.$$

Для вычисления  $y'(0,4)$  применим формулу (4.17):

$$y'(0,4) = (0,064 - 0,027) / 0,1 = 0,37$$

Для вычисления  $y'(0,3)$  используем более точную формулу (4.19)

$$y'(0,3) = (0,064 - 0,008) / (2 \cdot 0,1) = 0,28.$$

Для второй производной  $y''(0,3)$  применим (4.21)

$$y''(0,3) = (0,064 - 2 \cdot 0,027 + 0,008) / (0,1)^2 = 1,8.$$

**Примечание.** В данном примере таблично была задана функция  $y = x^3$ . В табл. 4.1 приведены сравнительные результаты точных и полученных приближенных значений. Анализ этих результатов показывает, во-первых, что вычисление  $y'(x)$  с помощью центральной разности (формула (4.19)) дает значительно более точные результаты, чем односторонние разности (формулы (4.16) и (4.17)), так

как у последних погрешность имеет первый порядок  $O(h)$ . Во-вторых, сравнение фактической погрешности аппроксимации вычисления производных показывает, что они отличаются незначительно. Небольшое отличие объясняется влиянием членов более высокого порядка малости, входящих в разложение функции в ряд Тейлора. В-третьих, полное совпадение вычисляемого и точного значения величины  $y''(0,3)$  объясняется тем, что погрешность аппроксимации для рассматриваемой функции  $y = x^3$  равна нулю, так как входящая в формулу погрешности аппроксимации величина  $f^{(4)}(x)$  и все производные более высокого порядка равны нулю (см. формулу (4.20)).

Таблица 4.1

### Результаты точных и полученных приближенных значений

$x_i$	$y_i'$ вычисленное	$y_i'$ точное	Фактическая погрешность	Погрешность аппроксимации
0,2	0,19	0,12	0,07	$y_i'' h/2 = 0,06$
0,3	0,28	0,27	0,01	$y_i''' h/6 = 0,01$
0,4	0,37	0,48	-0,11	$-(y_i'''' h)/2 = -0,12$

Окончание табл.4.1

$x_i$	$y_i''$ вычисленное	$y_i''$ точное	Фактическая погрешность	Погрешность аппроксимации
0,3	1,8	1,8	0	$-(y_i^{(4)} h^2)/12 = 0$

## 5. ЧИСЛЕННОЕ ИНТЕГРИРОВАНИЕ

### 5.1. Постановка задачи

Рассмотрим выражение

$$S = \int_a^b f(x) dx, \quad (5.1)$$

где  $f(x)$  – подынтегральная функция;  $a$  и  $b$  – нижний и верхний пределы интегрирования.

**Численное интегрирование** – это вычисление определенного интеграла (5.1) путем замены подынтегральной функции более простой

аппроксимирующей функцией, последующего прямого интегрирования и получения расчетных формул (квадратурные формулы). Идея численного интегрирования основывается на понятии геометрического смысла определенного интеграла, заключающегося в том, что значение определенного интеграла соответствует численному значению площади криволинейной трапеции, образованной при пересечении графиков функций  $y = f(x)$ ,  $x = a$ ,  $x = b$  и оси  $Ox$  (рис.5.1).

Рассмотрим сформулированное определение подробнее. Введем дискретное множество точек  $x_0, x_1, \dots, x_n$ , причем  $x_0 = a, x_n = b$ . Эти точки разбивают отрезок  $[a, b]$  на  $n$  элементарных отрезков  $[x_{i-1}, x_i]$  ( $i=1, 2, \dots, n$ ). Обозначим через  $h_i = x_i - x_{i-1}$  длину каждого элементарного отрезка. На каждом из этих отрезков выберем произвольную точку  $z_i (x_{i-1} \leq z_i \leq x_i)$  и найдем величину  $s_i$

$$s_i = f(z_i) \cdot h_i. \quad (5.2)$$

Составим сумму всех полученных величин  $s_i$  для всех элементарных отрезков

$$S_n = s_1 + s_2 + \dots + s_n = \sum_{i=1}^n f(z_i) h_i \quad (5.3)$$

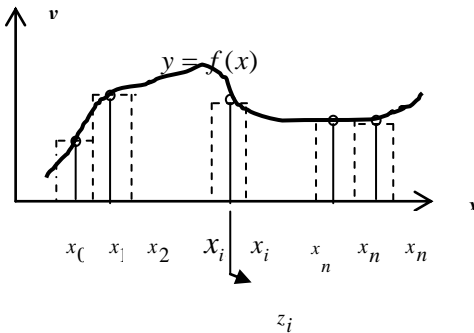


Рис.5.1. Геометрическая интерпретация определенного интеграла

Назовем интегральной суммой величину  $S_n$ . Обозначим через  $h = \max (h_i)$ . Тогда **определенным интегралом** от функции  $f(x)$  на отрезке  $[a, b]$  называется предел (если он существует) интегральной суммы при неограниченном увеличении числа точек разбиения, при котором длина наибольшего из отрезков стремится к нулю

$$\int_a^b f(x)dx = \lim_{h \rightarrow 0} S_n = \lim_{h \rightarrow 0} \sum_{i=1}^n f(z_i)h_i. \quad (5.4)$$

Отмеченные кружками точки (рис.5.1) на графике функции  $y = f(x)$  имеют абсциссы  $z_i$  и ординаты  $f(z_i)$ . Выражения вида (5.2) описывают площади элементарных прямоугольников (они отмечены штриховыми линиями), а интегральная сумма (5.3) – площадь ступенчатой фигуры, которая приближенно заменяет площадь криволинейной трапеции под кривой  $y = f(x)$ , ограниченной снизу осью  $Ox$ , а также прямыми  $x = x_0 = a$  и  $x = x_n = b$ . Если предел (5.4) существует при  $h \rightarrow 0$  и этот предел не зависит ни от способа разбиения на элементарные отрезки, ни от выбора точек  $z_i$ , то этот предел называется определенным интегралом от функции  $f(x)$  на отрезке  $[a, b]$ . Отсюда следует, что геометрический смысл определенного интеграла – это площадь описанной выше криволинейной трапеции, а сам интеграл является вещественным числом. Функция  $f(x)$ , для которой существует предел (5.4), называется интегрируемой. В частности, можно показать, что интегрируемой является любая непрерывная функция  $f(x)$ .

Различные формулы численного интегрирования, в сущности, основаны на соотношениях типа (5.3) и отличаются друг от друга следующим:

- выбором узловых точек  $x_i$ , и точек  $z_i$ , в которых вычисляется функция  $f(x)$ ;
- способом ускорения сходимости в процессе (5.4);
- величиной погрешностей и их оценкой, использующей дополнительную информацию о поведении функции  $f(x)$ .

Согласно курсу высшей математики известно, что если для функции  $f(x)$  можно аналитически найти первообразную функцию  $F(x)$  такую, что  $F'(x) = f(x)$ , то определенный интеграл можно вычислить по формуле Ньютона–Лейбница

$$\int_a^b f(x)dx = F(b) - F(a). \quad (5.5)$$

Однако на практике воспользоваться формулой (5.5) удастся далеко не всегда, и приходится использовать численные методы для вычисления определенного интеграла. Такая необходимость возникает в двух случаях:

1. Подынтегральная функция  $f(x)$  имеет такой вид, что первообразную  $F(x)$  нельзя выразить через элементарные функции с помощью приемов и методов, используемых в высшей математике. Примером такого интеграла является часто используемый в теории вероятности



интеграл вида  $S = \int_a^b e^{-x^2} dx$ , для которого нельзя в явном виде записать первообразную  $F(x)$ .

2. Подынтегральная функция  $f(x)$  задана в табличном виде, когда задано множество значений ее аргументов  $x_0, x_1, \dots, x_n$  и соответствующие им значения функции  $y_0, y_1, \dots, y_n$ .

Идея методов численного интегрирования основана на аппроксимации (приближении) подынтегральной функции  $f(x)$  какой-либо близкой к ней аппроксимирующей функцией  $\varphi(x)$ , интеграл от которой может быть вычислен аналитически, т.е. может быть найдена в явном виде первообразная этой функции. Удобнее всего для этой цели использовать многочленные аппроксимации  $\varphi(x) = P_n(x)$ , так как интегралы от многочленов вычисляются элементарно. Например, **квадратурные формулы Ньютона–Котеса** получаются путем замены подынтегральной функции интерполяционным многочленом Лагранжа с разбиением отрезка интегрирования на  $n$  равных частей. Часто встречающиеся на практике **квадратурные формулы прямоугольников, трапеций и Симпсона** – частные случаи формул Ньютона–Котеса, заменяющие подынтегральную функцию соответственно кусочно-постоянной, кусочно-линейной и кусочно-квадратичной функцией.

## 5.2. Метод прямоугольников

Это простейший метод численного интегрирования, в котором подынтегральная функция  $f(x)$  заменяется кусочно-постоянной функцией, т.е. функцией, постоянной на каждом отрезке  $[x_{i-1}, x_i]$ . Значение этой постоянной на каждом отрезке будет разным.

Геометрический смысл такой замены на одном элементарном отрезке  $[x_{i-1}, x_i]$  показан на рис. 5.2,а и, соответственно, на всем отрезке  $[a, b]$  на рис. 5.2,б. Видно, что искомая площадь заменяется суммой площадей соответствующих прямоугольников. На каждом элементарном отрезке

$$S_i = \int_{x_{i-1}}^{x_i} f(x) dx = f(x_{i-1} + h_i/2) \cdot h_i + R_i \quad (5.6)$$

а на всем отрезке  $[a, b]$

$$S = \int_a^b f(x) dx = \sum_{i=1}^n h_i \cdot f(x_{i-1} + h_i / 2) + R_i . \quad (5.7)$$

Если пренебречь погрешностью  $R_i$  на элементарном отрезке и погрешностью  $R$  на всем отрезке, то (5.6) даст квадратурную формулу на элементарном отрезке, а (5.7) – составную квадратурную формулу для метода прямоугольников.

Для применения этих квадратурных формул необходимо оценить величины погрешностей  $R_i$  и  $R$ . Будем считать, что  $f(x)$  непрерывна на  $[x_{i-1}, x_i]$  вместе со своей первой и второй производными. Вводя для функции  $f(x)$  ее первообразную  $F(x)$ , запишем левую часть (5.6) в виде:

$$\int_{x_{i-1}}^{x_i} f(x) dx = F(x_i) - F(x_{i-1}) . \quad (5.8)$$

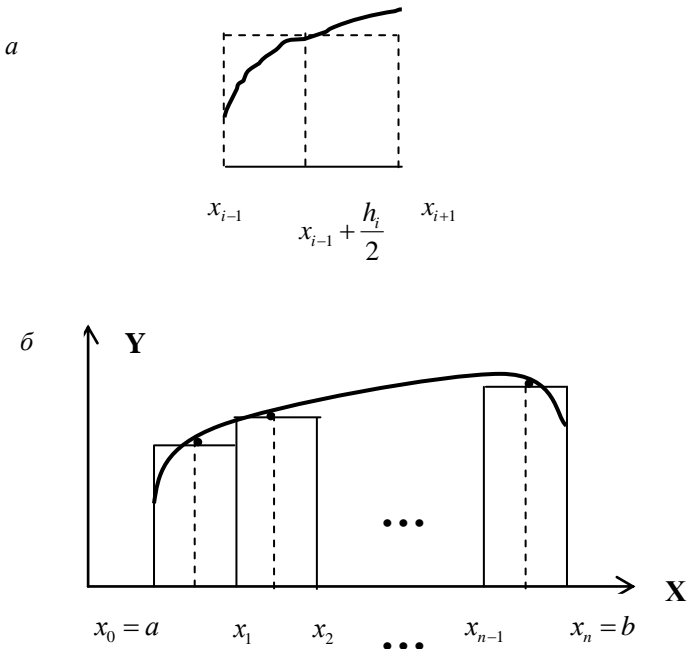


Рис. 5.2. Геометрический смысл замены подынтегральной функции.

Разлагая  $F(x)$  в ряд Тейлора в окрестности базовой точки  $x = z_i$  с остаточным членом в форме Лагранжа, получим значения функции  $F(x_i)$  и  $F(x_{i-1})$ :

$$F(x_i) = F(z_i) + \frac{h}{2} F'(z_i) + \left(\frac{h}{2}\right)^2 \frac{1}{2!} F''(z_i) + \left(\frac{h}{2}\right)^3 \frac{1}{3!} F'''(\xi_1); \quad (5.9)$$

$$F(x_{i-1}) = F(z_i) - \frac{h}{2} F'(z_i) + \left(\frac{h}{2}\right)^2 \frac{1}{2!} F''(z_i) - \left(\frac{h}{2}\right)^3 \frac{1}{3!} F'''(\xi_2); \quad (5.10)$$

где  $\xi_1 \in [z_i, x_i]$ , а  $\xi_2 \in [x_{i-1}, z_i]$ . Подставляя в правую часть (5.8) выражения (5.9) и (5.10) и учитывая, что  $F'(x) = f(x)$ , получим

$$\int_{x_{i-1}}^{x_i} f(x) dx = hf(z_i) + \frac{h^3}{24} f''(\xi), \quad (5.11)$$

где  $f''(\xi) = (f''(\xi_1) + f''(\xi_2))/2$  (такое значение  $\xi$  существует в силу непрерывности функции  $f(x)$ ). Последний член в (5.11) дает выражение для погрешности квадратурной формулы метода прямоугольников

$$R_i = \frac{h^3}{24} f''(\xi). \quad (5.12)$$

Для составной квадратурной формулы (5.7) вычисления интеграла на всем отрезке  $[a, b]$  погрешность имеет вид

$$R = \frac{h^2(b-a)}{24} f''(\xi), \xi \in [a, b]. \quad (5.13)$$

Видно, что для функции с ограниченной второй производной погрешности  $R_i$  и  $R$  убывают с уменьшением шага разбиения  $h$ . При малых  $h$   $R_i$  имеют третий порядок малости, а  $R$  – второй. Это объясняется тем, что при суммировании всех погрешностей  $R_i$  на всем отрезке  $[a, b]$  величина (5.12) умножается на количество элементарных отрезков  $n$ , которое при  $h = \text{const} = h_i$  связано с величиной  $h$  соотношением  $n = (b-a)/h$ .

### 5.3. Метод трапеций

Метод трапеций использует для описания функции  $f(x)$  линейную интерполяцию, заменяя ее кусочно-линейной функцией на отрезке  $[a, b]$ , т.е. ломаной, соединяющей точки  $x_i, y_i = f(x_i)$  (рис. 5.3).

В этом случае искомая площадь заменяется суммой трапеций, одна из которых приведена на рис. 5.3 *a*. Площадь этой трапеции легко вычисляется и дает квадратурную формулу

$$S_i = \int_{x_{i-1}}^{x_i} f(x) dx = \frac{1}{2}(y_{i-1} + y_i) \cdot h_i + R_i. \quad (5.14)$$

Методом, аналогичным приведенному выше, можно показать, что локальная погрешность метода трапеций имеет вид

$$R_i = -\frac{1}{12} h_i^3 f''(\xi_i), \xi_i \in [x_{i-1}, x_i] \quad (5.15)$$

а погрешность на всем интервале  $[a, b]$  равна

$$R = -\frac{h^2(b-a)}{12} f''(\xi), \xi \in [a, b] \quad (5.16)$$

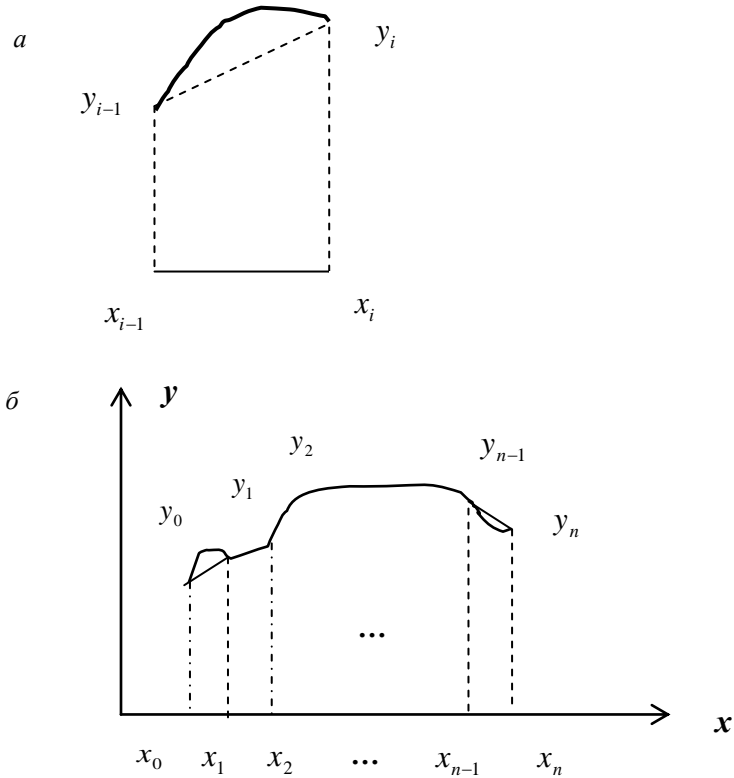


Рис.5.3. Геометрический смысл метода трапеций

Сравнение погрешностей  $R_i$  и  $R$  для метода трапеций (формулы (5.15) и (5.16)) и метода прямоугольников (формулы (5.12) и (5.13)) показывает, что эти погрешности имеют разные знаки, и погрешность метода прямоугольников в 2 раза меньше (с точностью до величины второй производной  $f''(\xi)$  в разных, вообще говоря, точках  $\xi$  для разных методов). Разные знаки у формул (5.13) и (5.16) указывают на то, что методы прямоугольников и трапеций дают двусторонние приближения к решению. Это означает, что если один из методов дает решение с недостатком, то другой метод – с избытком: точное решение находится в вилке между значениями, полученными методами прямоугольников и трапеций, что дает надежную оценку для величины определенного интеграла.

Квадратурная формула для метода трапеций имеет следующий вид:

$$S = \int_a^b f(x)dx = \sum_{i=1}^n \frac{1}{2}(y_{i-1} + y_i) \cdot h_i + R \quad (5.17)$$

В случае постоянного шага  $h_i = x_i - x_{i-1} = \text{const} = h$  и без учета погрешности формулу метода трапеций (17) можно записать в виде

$$\begin{aligned} S &= \frac{1}{2}h[(y_0 + y_1) + (y_1 + y_2) + (y_2 + y_3) + \dots + (y_{n-1} + y_n)] = \\ &= h\left[\frac{1}{2}(y_0 + y_n) + y_1 + y_2 + \dots + y_{n-1}\right] \end{aligned} \quad (5.18)$$

Рассмотрим применение метода прямоугольников и трапеций на примере.

**Пример 5.1.** Вычислить интеграл

$$S = \int_0^1 \frac{dx}{1+x^2}.$$

*Решение.* Этот интеграл является табличным и легко вычисляется по формуле Ньютона – Лейбница

$$\bar{S} = \arctg(x)\Big|_0^1 = \pi/4 = 0,785398.$$

Применим для вычисления интеграла метод прямоугольников (формула (5.7)) и метод трапеций (формула (5.14)). Разобьем отрезок интегрирования  $[0, 1]$  на пять равных частей:  $n = 5, h = 0,2$ . Вычислим значения подынтегральной функции  $y_i = 1/(1+x_i^2)$  в точках разбиения  $x_i = x_{i-1} + 0,2$  (для метода трапеций) и в серединах отрезков разбиения  $z_i = x_{i-1} + 0,5h$  (для метода прямоугольников). Результаты вычислений представлены в табл. 5.1.

Таблица 5.1

## Результаты вычисления подынтегральной функции

$x_i$	$y_i = y(x)$	$z_i = x_{i-1} + 0,5h$	$y(z_i)$	Метод
0,0	1,000000	0,1	0,990099	Прямоугольников $S_{II} = 0,786237$
0,2	0,961538	0,3	0,917431	
0,4	0,862069	0,5	0,800000	Трапеций $S_T = 0,783731$
0,6	0,735294	0,7	0,671141	
$x_i$	$y_i = y(x)$	$z_i = x_{i-1} + 0,5h$	$y(z_i)$	Метод
0,8	0,609756	0,9	0,552486	Точное решение $\bar{S} = 0,785398$
1,0	0,500000			

Методом прямоугольников получим

$$S_{II} = 0,2(0,990099 + 0,917431 + 0,8 + 0,67114 + 0,552486) = 0,2 \cdot 3,931157 = 0,786231.$$

Методом трапеций получим

$$S_T = 0,2((1 + 0,5) \cdot 0,5 + 0,961538 + 0,862069 + 0,735294 + 0,609756) = 0,2 \cdot 3,918657 = 0,783731.$$

#### 5.4. Метод Симпсона

Во многих случаях требуется вычислять интегралы для таблично заданных функций, когда точки  $x_i$  разбиения отрезка  $[a, b]$  на элементарные отрезки заданы и повысить точность численного интегрирования путем уменьшения шага  $h_i$  не удастся. В этом случае желательно использовать квадратурные формулы, которые при том же разбиении отрезка дают более высокую точность, чем методы прямоугольников и трапеций. Одним из таких методов является метод Симпсона. Повышение точности в нем достигается за счет повышения точности интерполяции – функция  $f(x)$  заменяется интерполяционным многочленом Лагранжа второй степени (квадратичная интерполяция).

Разобьем отрезок интегрирования  $[a, b]$  на четное число равных частей, полагая  $x_{i+1} - x_i = x_i - x_{i-1} = h$ . Рассмотрим два соседних интервала  $[x_{i-1}, x_i]$  и  $[x_i, x_{i+1}]$  (рис. 5.4).

Проведем через три точки  $(x_{i-1}, y_{i-1})$ ,  $(x_i, y_i)$ ,  $(x_{i+1}, y_{i+1})$  интерполяционный многочлен Лагранжа второй степени

$$L_2(x) = \frac{(x-x_1)(x-x_2)}{(x_0-x_1)(x_0-x_2)} y_0 + \frac{(x-x_0)(x-x_2)}{(x_1-x_0)(x_1-x_2)} y_1 + \frac{(x-x_0)(x-x_1)}{(x_2-x_0)(x_2-x_1)} y_2. \quad (5.19)$$

Вычислим определенный интеграл на отрезке  $[x_{i-1}, x_{i+1}]$ , заменив подынтегральную функцию  $f(x)$  многочленом  $L_2(x)$ ; проинтегрировав, получим

$$S_i = \int_{x_{i-1}}^{x_{i+1}} f(x) dx \approx \int_{x_{i-1}}^{x_{i+1}} L_2(x) dx \approx \frac{h}{3} (y_{i-1} + 4y_i + y_{i+1}). \quad (5.20)$$

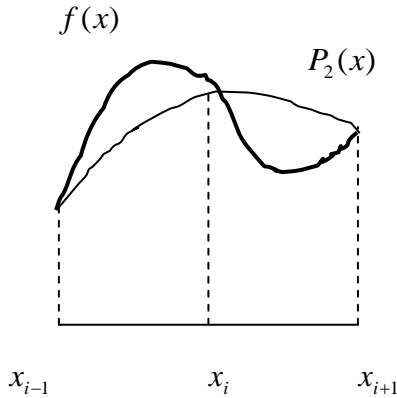


Рис. 5.4. Рассматриваемые интервалы

$[x_0, x_2]$ ,  $[x_2, x_4]$ , ...,  $[x_{n-2}, x_n]$ , можно получить составную квадратурную формулу метода Симпсона

$$S = \int_a^b f(x) dx \approx \frac{h}{3} [(y_0 + 4y_1 + y_2) + (y_2 + 4y_3 + y_4) + \dots + (y_{n-2} + 4y_{n-1} + y_n)] = \frac{h}{3} [y_0 + 4y_1 + 2y_2 + 4y_3 + 2y_4 + \dots + 2y_{n-2} + 4y_{n-1} + y_n].$$

В этой формуле члены в квадратных скобках сгруппированы следующим образом: крайние члены ( $y_0 = f(a)$  и  $y_n = f(b)$ ) входят в сумму с коэффициентом, равным единице; члены с четными номерами ( $y_2, y_4, \dots, y_{n-2}$ ) умножаются на два, а члены с нечетными номерами

$(y_1, y_3, \dots, y_{n-1})$  умножаются на четыре. Исследование погрешности метода Симпсона показывает, что локальная погрешность на отрезке  $[x_{i-1}, x_{i+1}]$  имеет вид

$$R_i = \frac{h^5}{90} f^{(4)}(\xi_i) \quad (\xi \in [x_{i-1}, x_{i+1}]), \quad (5.21)$$

а погрешность на всем отрезке  $[a, b]$

$$R = \frac{h^4(b-a)}{180} f^{(4)}(\xi) \quad (\xi \in [a, b]) \quad (5.22)$$

где  $f^{(4)}(\xi)$  – четвертая производная функции  $f(x)$  в некоторой точке  $x = \xi$ . Видно, что для достаточно гладких функций, когда  $f^{(4)}(x)$  ограничена, метод Симпсона является гораздо более точным, чем методы прямоугольников и трапеций. Вычислим с его помощью определенный интеграл, рассмотренный в примере 4.1.

**Пример 5.2.** Вычислить методом Симпсона интеграл

$$S = \int_0^1 \frac{dx}{1+x^2}.$$

*Решение.* Разобьем отрезок  $[0, 1]$  на четыре элементарных отрезка:  $n = 4$ ,  $h = 0,25$ . Результаты вычисления подынтегральной функции  $y_i = 1/(1+x_i^2)$  приведены в табл. 5.2.

Таблица 5.2

**Результаты вычисления подынтегральной функции  $y_i = 1/(1+x_i^2)$**

$x_i$	$y_i$	Точное решение $\bar{S} = 0,785398$
0,0	1,000000	
0,25	0,941176	Метод Симпсона $S_c = 0,785392$
0,50	0,800000	
0,75	0,640000	Погрешность $\bar{S} - S_c = 0,000006$
1,0	0,500000	

Сравнение с результатами, приведенными в табл. 5.1, показывает, что хотя шаг  $h$  в данном случае выбран даже несколько больше ( $h = 0,25$  вместо  $h = 0,2$  в примере 5.1), погрешность метода Симпсона в 278 раз меньше, чем погрешность метода трапеций.



## 6. ЧИСЛЕННЫЕ МЕТОДЫ РЕШЕНИЯ НЕЛИНЕЙНЫХ УРАВНЕНИЙ И СИСТЕМ НЕЛИНЕЙНЫХ УРАВНЕНИЙ

### 6.1. Нелинейные уравнения

#### 6.1.1. Постановка задачи

Рассмотрим уравнение вида

$$F(x) = 0, \quad (6.1)$$

где  $F(x)$  – непрерывная функция достаточно произвольного вида, определенная на некотором конечном или бесконечном интервале.

**Корнем уравнения** (6.1) называется всякое число  $x^*$ , обращающее в нуль функцию  $F(x)$ , т. е. такое, что

$$F(x^*) = 0. \quad (6.2)$$

Задача нахождения всех корней уравнения (6.1) обладает следующими характерными особенностями:

1. В общем случае **не существует “конечных” алгоритмов** для получения корней уравнения (6.1), т. е. таких алгоритмов, которые позволяют за конечное число арифметических операций получить точное решение.

Исключением являются уравнения типа

$$\sin x = 0, \quad (6.3)$$

некоторые простые многочлены вида

$$ax(bx + c) = 0, \quad (6.4)$$

и другие подобные уравнения, в которых решение может быть сразу выписано или без труда найдено с помощью небольшого числа арифметических операций.

Даже в простом случае квадратного уравнения решение выписывается с помощью радикалов (квадратных корней), вычисление которых выполняется приближенно. Показано, что для многочленов выше четвертой степени точное решение не может быть получено даже с помощью радикалов. Поэтому обычно используются приближенные методы решения (6.1) – графические или численные.

Численные методы решения (6.1) являются, как правило, итерационными. Идея этих методов заключается в том, что, задавшись некоторым приближенным начальным значением  $x_0$  для корня уравнения, можно с помощью некоторой совокупности операций найти уточненное значение корня. Выполнение этой совокупности операций определяет одну итерацию. Многократное повторение итераций, образующее итерационный процесс, позволяет получать все более точное значение корня (при условии, что итерационный процесс сходится). Таким образом, **итерационный процесс** заключается в вычислении последова-

тельности значений  $x_1, x_2, \dots, x_k, \dots$ , где величина  $k$  указывает номер итерации или номер приближения. **Итерационный процесс сходится** к корню  $x^*$  уравнения, если  $\lim_{k \rightarrow \infty} x_k = x^*$ .

**2. Количество корней уравнения (6.1) обычно заранее неизвестно.**

При различных  $F(x)$  уравнение (6.1) может вообще не иметь корней, иметь один, несколько или даже бесчисленное множество корней. Например, уравнение  $x^2 + 5 = 0$  не имеет действительных корней, уравнение  $e^x + x = 0$  имеет единственный корень, уравнение (6.4) – два корня (при  $b \neq 0$ ), уравнение (6.3) – бесчисленное множество корней. Поэтому численному решению (6.1) должен предшествовать анализ уравнения и выяснение того, сколько корней оно имеет. Это удобнее всего сделать методами математического анализа.

Кроме того, необходимо **отделить все корни**, т.е. для каждого корня  $x^*$  указать такой интервал  $[a, b]$ , в котором расположен данный корень и других корней в нем нет. При использовании итерационных методов предполагается, что отделение корней проведено, так что итерационный процесс используется для приближенного определения единственного корня на заданном интервале.

**3. Сходимость большинства итерационных методов зависит от начального приближения  $x_0$ .** Как правило, сходимость гарантируется только в том случае, если  $x_0$  достаточно близко к корню уравнения  $x^*$ . Иногда полезно использовать понятие области притяжения корня. **Областью притяжения корня  $x^*$**  для данного итерационного метода называется такое множество  $A$  значений величины  $x_0$ , что для любого  $x_0 \in A$  итерационный метод сходится к корню  $x^*$ .

При сравнении сходимости различных методов часто используется понятие **порядка сходимости метода**. Говорят, что метод имеет  $n$ -й порядок сходимости, если

$$|x_k - x^*| \leq A|x_{k-1} - x^*|^n, \quad (6.5)$$

где  $A$  – константа, зависящая от вида функции  $F(x)$ , но не зависящая от номера итерации  $k$ . Неравенство (6.5) фактически связывает погрешности на двух последовательных итерациях и показывает, насколько быстро сходится итерационный процесс. Далее будут рассмотрены методы, имеющие линейную сходимость

$$|x_k - x^*| \leq A|x_{k-1} - x^*| \quad (6.6)$$

и квадратичную сходимость

$$|x_k - x^*| \leq A|x_{k-1} - x^*|^2 \quad (6.7)$$

Величина  $A$  для каждого метода различна и зависит, кроме того, от вида функции  $F(x)$ .

### 6.1.2. Отделение корней нелинейного уравнения

Рассмотрим пример отделения корней.

**Пример 6.1.** Определить количество корней уравнения  $e^x - x - 2 = 0$  и отделить корни.

*Решение.* Запишем уравнение в виде  $e^x = x + 2$  и построим графики функций, стоящих в левой и правой части уравнения  $y_1 = e^x$  и  $y_2 = x + 2$  (рис. 6.1).

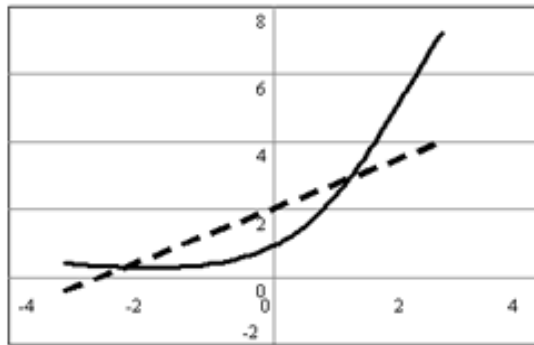


Рис. 6.1. Графики функций, стоящих в левой и правой части уравнения  $y_1 = e^x$  и  $y_2 = x + 2$ :

—————  $y_1(x) = \exp(x)$     - - - - -  $y_2(x) = x + 2$

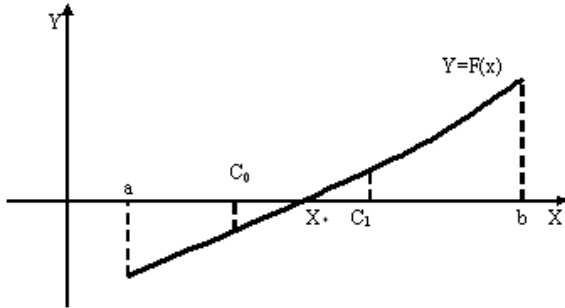
Корнями уравнения будут те значения  $x$ , при которых эти графики пересекаются. При  $x = 1$  график прямой лежит выше графика функции  $y_1 = e^x$ , так как при  $x = 1$   $y_1 = e^1 = e \approx 2,71$ , а  $y_2 = 1 + 2 = 3$ . При  $x = 2$  прямая расположена ниже графика  $y_1 = e^x$ , так как  $e^2 \approx 7,4 > 2 + 2 = 4$ . Следовательно, один из корней находится в интервале  $[1, 2]$ . Вычислив значения функций  $y_1 = e^x$  и  $y_2 = x + 2$  при  $x = -2$  и  $x = -1$ , определяем, что второй корень (отрицательный)

находится внутри интервала  $[-2, -1]$ . Зная поведение функций  $y_1 = e^x$  и  $y_2 = x + 2$ , мы из рис. 6.1 можем заключить, что других корней у рассматриваемого уравнения нет. Функция  $y_1 = e^x$  с ростом  $x$  растет гораздо быстрее, чем прямая  $y_2 = x + 2$  и при  $x > 2$  других корней быть не может. Аналогично обстоит дело при  $x < -2$ , так как  $e^x \rightarrow 0$  при  $x \rightarrow -\infty$ , а  $y_2 = x + 2 \rightarrow -\infty$  при  $x \rightarrow -\infty$ . Из этого рассмотрения следует, что положительный корень уравнения надо искать на интервале  $[1, 2]$ , а отрицательный – на интервале  $[-2, -1]$ , причем положительный корень ближе к точке  $x = 1$ , а отрицательный – к точке  $x = -2$ .

### 6.1.3. Метод половинного деления и метод хорд

**Метод половинного деления** – всегда сходящийся для непрерывной функции метод нахождения корня. Предположим, что корни уравнения (6.1) уже отделены, и на заданном интервале  $[a, b]$  уравнение имеет единственный корень  $x^*$ , и, следовательно, на концах интервала функция имеет разные знаки:  $F(a) \cdot F(b) < 0$ . Никаких ограничений на функцию  $F(x)$ , кроме требования непрерывности, не накладывается. Идея метода заключается в том, чтобы вместо исходного интервала  $[a, b]$  получить путем последовательного деления отрезка пополам достаточно малый интервал  $[a, b]$ , внутри которого находится корень.

Разделим интервал  $[a, b]$  пополам, вычислив координату середины отрезка  $c_0 = (a+b)/2$ . В результате получим два интервала:  $[a, c_0]$  и  $[c_0, b]$ . Если окажется, что  $F(c_0) = 0$ , то значит  $c_0$  является корнем ( $x^* = c_0$ ) и, следовательно, решение на этом заканчивается. Если же  $F(c_0) \neq 0$ , то из двух интервалов выбираем тот, на концах которого функция  $F(x)$  имеет разные знаки и внутри которого, следовательно, находится корень. Второй интервал, на котором нет корня, отбрасывается. Для функции  $F(x)$ , приведенной на рис. 6.2,  $F(c_0) \cdot F(a) > 0$  и поэтому  $F(c_0) \cdot F(b) < 0$ , откуда следует, что надо оставить отрезок  $[c_0, b]$ , а отрезок  $[a, c_0]$  отбросить, так как  $x^* \in [c_0, b]$ .

Рис.6.2 Функция  $F(x)$ 

Если обозначить координаты оставленного интервала через  $a_1$  и  $b_1$ , то мы получим, что в общем случае  $a_1 = a$ ,  $b_1 = c_0$ , если  $F(c_0) \cdot F(a) < 0$ , и  $a_1 = c_0$ ,  $b_1 = b$ , если  $F(c_0) \cdot F(b) < 0$ . Длина полученного интервала  $[a_1, b_1]$ , внутри которого находится корень  $x^*$ , будет вдвое меньше длины исходного интервала  $[a, b]$ . На этом заканчивается один шаг метода. Разделим теперь полученный интервал  $[a_1, b_1]$  снова пополам, вычислив  $c_1 = (a_1 + b_1) / 2$ . Из полученных двух интервалов  $[a_1, c_1]$  и  $[c_1, b_1]$  возьмем тот, на концах которого  $F(x)$  имеет разные знаки. Продолжая этот процесс, мы получим последовательность вложенных интервалов  $[a_1, b_1]$ ,  $[a_2, b_2]$ , ...,  $[a_k, b_k]$ , ..., внутри которых находится корень и длина которых монотонно убывает, так как

$$b_k - a_k = \frac{1}{2^k} (b - a). \quad (6.8)$$

Процесс вычислений прекращается, когда  $b_k - a_k < \varepsilon$ , где  $\varepsilon$  – требуемая точность решения задачи. Таким образом, в методе половинного деления находится не значение корня, а малый интервал  $[a_k, b_k]$ , в котором находится этот корень.

Главное достоинство метода заключается в том, что он сходится всегда, независимо от того, как ведут себя функция  $F(x)$  и ее производные. Единственное требование – наличие разных знаков на концах интервала (и, разумеется, непрерывность  $F(x)$ ). Если на интервале имеется не единственный корень, то метод позволяет получить один из корней. Недостатком метода является сравнительно медленная сходимость.

**Пример 6.2.** Найти методом половинного деления положительный корень уравнения

$$f(x) \equiv e^x - x - 2 = 0 \text{ с точностью } \varepsilon = 0,02.$$

*Решение.* В предыдущем примере было показано, что этот корень находится внутри интервала  $[1, 2]$ . Приведем полученные на различных шагах метода результаты в виде табл. 6.1.

*Ответ:*  $x \in [1,1406...1,15625]$ .

Используя основную идею метода половинного деления (разбиение исходного отрезка на два и отбрасывание того отрезка, где нет корня), можно модифицировать метод, деля отрезок не пополам, а в каком либо другом отношении. Одной из таких модификаций является метод хорд, существо которого заключается в следующем (рис.6.3, 6.4).

Таблица 6.1

**Результаты решения уравнения  $f(x) \equiv e^x - x - 2 = 0$**

**методом половинного деления**

$k$	$a_k$	$b_k$	$c_k$	$f(a_k)$	$f(b_k)$	$f(c_k)$	$b_k - a_k$
0	1	2	1,5	-0,2817	3,3891	0,9817	1
1	1	1,5	1,25	-0,2817	0,9817	0,2403	0,5
2	1	1,25	1,125	-0,2817	0,2403	-0,0448	0,25
3	1,125	1,25	1,1875	-0,0448	0,2403	0,0914	0,125
4	1,125	1,1875	1,15625	-0,0448	0,0914	0,0217	0,0625
5	1,125	1,15625	1,1606	-0,0448	0,0217	-0,0188	0,03125
6	1,1406	1,15625					0,01565

В качестве точки  $c_0$ , делящей отрезок  $[a, b]$  на две части, выбирается точка пересечения с осью  $Ox$  прямой  $AB$ , проведенной через две точки  $A(a, F(a))$  и  $B(b, F(b))$ . Получив два отрезка  $[a, c_0]$  и  $[c_0, b]$ , мы оставляем тот из них, на концах которого функция  $F(x)$  имеет разные знаки и на котором, следовательно, находится корень. На рис. 6.3 это отрезок  $[a, c_0]$ , для которого  $F(c_0) \cdot F(a) < 0$ , а на рис. 6.4 – отрезок  $[c_0, b]$ , для которого  $F(c_0) \cdot F(b) < 0$ . Прямая  $AB$  будет хордой, соединяющей две крайние точки кривой  $F(x)$ . На следующем шаге метода проводится хорда через крайние точки функции  $F(x)$  на отрезке  $[a, c_0]$  (или  $[c_0, b]$  на рис. 6.4) и определяется точка  $c_1$  пересечения этой хорды с осью  $Ox$ . Для функции  $F(x)$ , приведенной на рис. 6.3, видно, что все приближения метода хорд  $c_0, c_1, c_2, \dots, c_k$  лежат правее корня

$x^* (c_k > x^*)$  и итерационный процесс будет монотонным. Аналогично обстоит дело для функции, приведенной на рис. 6.4, но теперь последовательность  $c_0, c_1, c_2, \dots, c_k$  будет лежать левее корня. Итерационный процесс продолжается до тех пор, пока значение  $F(c_k)$  не станет по модулю меньше заданного малого числа  $\varepsilon$ . Значения величин  $c_0, c_1, c_2, \dots, c_k$  определяются следующим образом. Уравнение прямой  $AB$ , проходящей через точки  $A(a, F(a))$  и  $B(b, F(b))$  имеет вид

$$\frac{x-a}{b-a} = \frac{y-F(a)}{F(b)-F(a)}.$$

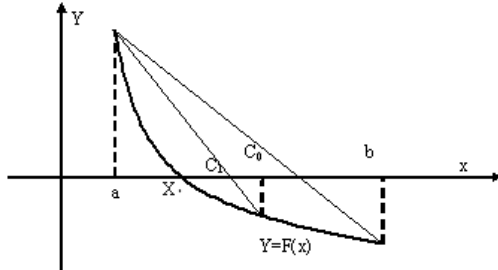


Рис. 6.3. Отрезок  $[a, c_0]$ , для которого  $F(c_0) \cdot F(a) < 0$

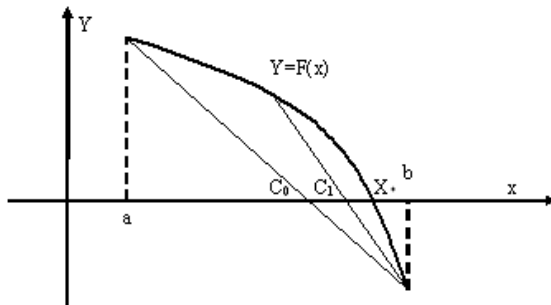


Рис. 6.4. Отрезок  $[c_0, b]$ , для которого  $F(c_0) \cdot F(b) < 0$

Для точки ее пересечения с осью  $Ox$  ( $x = c_0, y = 0$ ) получим уравнение

$$\frac{c_0 - a}{b - a} = \frac{-F(a)}{F(b) - F(a)},$$

откуда для  $c_0$  получим выражение

$$c_0 = a - \frac{b-a}{F(b)-F(a)} F(a). \quad (6.9)$$

Аналогично определяется следующее приближение  $c_1$  как пересечение с осью  $Ox$  хорды для  $F(x)$  на отрезке  $[a, c_0]$  ( $[c_0, b]$  для случая на рис. 6.4). Для функции  $F(x)$ , приведенной на рис. 6.3 и 6.4, первая и вторая производные  $F'(x)$  и  $F''(x)$  не обращаются в нуль ни в одной точке интервала  $[a, b]$ . В этом случае сходимость метода хорд будет монотонной. На самом деле применение метода хорд в том виде, как он здесь описан, не накладывает никаких ограничений на функцию  $F(x)$ , кроме ее непрерывности и условия  $F(a) \cdot F(b) < 0$ , достаточных для существования хотя бы одного корня на отрезке  $[a, b]$ . Как и метод половинного деления, **метод хорд сходится всегда**, однако в ряде случаев он дает более быструю сходимость итерационного процесса, чем метод половинного деления. Такая ситуация является типичной для нелинейных задач. **Сходимость и сравнительные достоинства различных методов могут существенно зависеть от поведения функции  $F(x)$  и ее производных.**

Завершая рассмотрение двух всегда сходящихся методов, отметим одно несомненное достоинство метода половинного деления: для него всегда **заранее можно сказать, сколько шагов потребуется для получения корня с заданной точностью** (независимо от вида функции  $F(x)$ ). Эта оценка делается на основании формулы (6.8). Метод хорд таким свойством не обладает, так как скорость его сходимости зависит от вида функции  $F(x)$ .

**Пример 6.3.** Найти методом хорд положительный корень уравнения  $F(x) \equiv e^x - x - 2 = 0$  с точностью  $\varepsilon = 0,01$ .

*Решение.* В предыдущем рассмотрении этого примера было показано, что этот корень находится в интервале  $[1, 2]$ . Примем  $a = 1$ ,  $b = 2$  и проведем несколько итераций методом хорд по формуле (6.9). Результаты расчетов приведены в табл. 6.2. Сравнение этих результатов с результатами метода половинного деления (табл. 6.1) показывает, что для уравнения  $e^x - x - 2 = 0$  метод хорд сходится быстрее.

#### 6.1.4. Метод простой итерации

Для того, чтобы применить метод простой итерации, необходимо прежде всего привести уравнение (6.1) к виду, удобному для проведения итерационного процесса

$$x = \varphi(x), \quad (6.10)$$



причем уравнения (6.10) и (6.1) должны быть равносильны, т.е. все корни одного уравнения должны быть корнями другого уравнения, и наоборот.

Таблица 6.2

**Сравнение результатов метода хорд и метода  
половинного деления**

$k$	$a_k$	$b_k$	$c_k$	$F(a_k)$	$F(b_k)$	$F(c_k)$	интервал	$ c_k - c_{k-1} $
0	1	2	1,6216	-0,2817	3,3891	1,4395	$[a_0, c_0]$	
1	1	1,621	1,1017	-0,2817	1,4395	-0,09236	$[c_1, b_1]$	0,5199
2	1,102	1,621	1,1331	-0,0924	1,4395	-0,0280	$[c_2, b_2]$	0,0321
3	1,134	1,621	1,1424	-0,0280	1,4395	-0,00822	$[c_3, b_3]$	0,0093

Привести (6.1) к виду (6.10) можно различными способами. Рассмотрим пример.

**Пример 6.4.** Привести уравнение

$$x^3 + x - 0,5 = 0 \quad (6.11)$$

к виду, удобному для итераций (6.10).

*Решение.* Если перенести все члены, кроме второго, в правую часть, получится равносильное уравнение

$$x = 0,5 - x^3. \quad (6.12)$$

Если перенести второй и третий члены в правую часть и извлечь из обеих частей кубический корень, получится уравнение

$$x = \sqrt[3]{0,5 - x}. \quad (6.13)$$

Уравнения (6.12)–(6.13) равносильны (6.11) и имеют вид (6.10), где в качестве  $\varphi(x)$  будут соответственно  $\varphi_1(x) \equiv 0,5 - x^3$ ,  $\varphi_2(x) \equiv \sqrt[3]{0,5 - x}$ . Можно указать еще множество других способов сведения уравнения (6.11) к виду (6.10). Метод простой итерации для уравнения  $x = \varphi(x)$  заключается в следующем.

Зададимся начальным приближением  $x_0$  и построим последовательность чисел  $x_1, x_2, \dots, x_k, \dots$  с помощью формул

$$x_1 = \varphi(x_0), x_2 = \varphi(x_1), \dots, x_k = \varphi(x_{k-1}), \dots \quad (6.14)$$

Если эта последовательность сходится, то для непрерывной функции  $\varphi(x)$  она сходится к корню уравнения (6.10). Действительно,

пусть  $\lim_{k \rightarrow \infty} x_k = d$ . Тогда  $d = \lim_{k \rightarrow \infty} x_k = \lim_{k \rightarrow \infty} \varphi(x_{k-1}) = \varphi(\lim_{k \rightarrow \infty} x_k) = d$ . Отсюда следует, что  $d$  – корень уравнения (6.10).

Достаточное условие сходимости метода простой итерации определяется следующей теоремой.

**Теорема.** Пусть  $x^*$  – корень уравнения (6.10), т.е.

$$x^* = \varphi(x^*) . \quad (6.15)$$

Пусть функция  $\varphi(x)$  определена и дифференцируема на отрезке  $[a, b]$  и все элементы последовательности (5.21) также принадлежат этому отрезку. Пусть существует действительное число  $q$  такое, что

$$|\varphi'(x)| \leq q < 1, x \in [a, b]. \quad (6.16)$$

Тогда процесс итераций (6.14) сходится и предельное значение  $x^* = \lim_{k \rightarrow \infty} x_k$  является единственным корнем уравнения (6.10).

**Доказательство.** Оценим по формуле Лагранжа погрешность на двух последовательных итерациях. В силу (6.14) и (6.16) и ввиду того, что  $x_k \in [a, b]$ ,

$$x_k - x^* = \varphi(x_{k-1}) - \varphi(x^*) = \varphi'(\zeta) \cdot (x_k - x^*), \zeta \in [x_k, x^*]$$

откуда в силу (6.16)

$$|x_k - x^*| = q |x_{k-1} - x^*| . \quad (6.17)$$

Применяя (6.17) для  $k = 1$ , получим

$$|x_1 - x^*| = q |x_0 - x^*| .$$

Для  $k = 2$ , используя предыдущее неравенство, имеем

$$|x_2 - x^*| \leq q |x_1 - x^*| \leq q^2 |x_0 - x^*| .$$

С помощью метода математической индукции легко показать, что в общем случае

$$|x_k - x^*| \leq q^k |x_0 - x^*| . \quad (6.18)$$

Так как  $q < 1$ , то из (6.18) следует, что последовательность  $x_k$  сходится к корню уравнения  $x^* = \lim_{k \rightarrow \infty} x_k$ , так как  $\lim_{k \rightarrow \infty} q^k = 0$ . Скорость этой сходимости не хуже, чем у геометрической прогрессии  $q^k$  ( $q < 1$ ).

Покажем, что корень  $x^*$  является единственным. Действительно, пусть существует другой корень  $x^{**}$ , так что  $x^{**} = \varphi(x^{**})$ ,  $x^{**} \in [a, b]$ . Тогда по формуле Лагранжа

$$x^* - x^{**} = \varphi(x^*) - \varphi(x^{**}) = \varphi'(\zeta) \cdot (x^* - x^{**}), \zeta \in [x^*, x^{**}] \quad \text{откуда}$$

$$|x^* - x^{**}| \leq q |x^* - x^{**}| < |x^* - x^{**}| .$$

Полученное противоречие доказывает единственность корня.

**Пример 6.5.** Исследовать применение метода итераций на примере уравнения (6.11)

$$x^3 + x - 0,5 = 0.$$

*Решение.* Построив функцию  $y = x^3$  и  $y = 0,5 - x$  графически, легко определить, что корень принадлежит интервалу  $[0,3; 0,5]$ . Как было показано выше, привести данное уравнение к виду, удобному для итераций, можно различными способами. Ниже приведены эти уравнения вместе с вычисленными производными от правых частей

$$x = 0,5 - x^3 \equiv \varphi_1(x); \varphi_1'(x) = -3x^2; \quad (6.19)$$

$$x = \sqrt[3]{0,5 - x} \equiv \varphi_2(x); \varphi_2'(x) = \frac{1}{3} \cdot \frac{1}{\sqrt[3]{(0,5 - x)^2}}. \quad (6.20)$$

Из (6.19) и (6.20) видно, что при  $x_k \in [0,3..0,5]$   $|\varphi_1'(x_k)| < 1$ , а  $|\varphi_2'(x_k)| > 1$ . Следовательно, метод простой итерации для (6.19) сходится, а для (6.20) – расходится, если взять  $x_k \in [0,3..0,5]$ . В табл. 6.3 приведены результаты расчета корня данного уравнения на различных итерационных шагах.

Таблица 6.3

### Результаты расчета корня уравнения

$k$	$x_k$	$ x_k - x^* $
0	0,5	0,076146
3	0,410526	0,013328
6	0,425890	0,002036
9	0,423534	0,000320
12	0,423904	0,000050
15	0,423847	0,000007

### 6.1.5. Метод Ньютона

**Метод Ньютона (касательных)** – быстро сходящийся итерационный метод нахождения корня, сходимость которого сильно зависит от выбора начального приближения.

Пусть  $x^*$  – корень уравнения (6.1),  $x^* \in [a, b]$ . Допустим, что существуют  $F'(x), F''(x)$ , причем они не меняют знак на отрезке  $[a, b]$

$$F'(x) \cdot F''(x) \neq 0 \quad (6.21)$$

Зададимся начальным приближением  $x_0 \in [a, b]$ , причем будем считать, что оно выбрано таким образом, что выполняется условие

$$F(x_0) \cdot F''(x_0) > 0. \quad (6.22)$$

Совокупность условий (6.21) и (6.22) называется **условиями Фурье**. Условие (6.21) фактически означает, что  $F'(x)$  и  $F''(x)$  будут знакопостоянными функциями.

Будем искать корень (6.21) в виде

$$x^* = x_0 + h. \quad (6.23)$$

Воспользуемся разложением в ряд Тейлора функции  $F(x)$  в окрестности точки  $x_0$

$$F(x^*) = F(x_0 + h) = F(x_0) + F'(x_0) \cdot h + \frac{1}{2} F''(\zeta) \cdot h^2. \quad (6.24)$$

Будем считать, что величина  $x_0$  достаточно близка к корню  $x^*$  и, следовательно,  $h$  достаточно мало. Тогда, пренебрегая членом порядка  $h^2$  в разложении и учитывая, что  $F(x^*) = 0$ , получим

$$F(x_0) + F'(x_0) \cdot h = 0,$$

откуда можно найти

$$h = -F(x_0) / F'(x_0). \quad (6.25)$$

Так как в (6.24) отброшен член, содержащий  $h^2$ , то (6.23) дает не величину  $x^*$ , а уточненное значение корня  $x_1$

$$x_1 = x_0 - F(x_0) / F'(x_0).$$

Продолжая этот процесс, получаем следующую итерационную формулу метода Ньютона

$$x_{k+1} = x_k - F(x_k) / F'(x_k). \quad (6.26)$$

Метод Ньютона имеет простую геометрическую интерпретацию: полученное по формуле (6.26) значение есть точка пересечения касательной к кривой  $y = F(x)$  в точке  $x_k$  с осью абсцисс. Действительно, уравнение этой касательной имеет вид

$$y - F(x_k) = F'(x_k)(x - x_k),$$

откуда при  $y = 0$  находим пересечение с осью абсцисс  $x = x_k - F(x_k) / F'(x_k)$ , что в точности совпадает с  $x_{k+1}$ , получаемым в методе Ньютона по (6.26). Поэтому метод Ньютона называют иногда методом касательных.

### 6.1.6. Метод секущих

В методе Ньютона требуется вычисление  $F(x_k)$  и  $F'(x_k)$ . Но иногда производная  $F'(x_k)$  вычисляется довольно сложно и желательно ограничиться только вычислением функции  $F(x)$ . Существуют методы, которые позволяют по двум приближениям  $x_k$  и  $x_{k-1}$  построить следующее приближение  $x_{k+1}$ . Идея заключается в том, чтобы на кривой  $F(x)$  через две точки  $(x_k, F(x_k))$  и  $(x_{k-1}, F(x_{k-1}))$  провести прямую, и точку пересечения этой прямой с осью  $Ox$  считать следующим приближением. Обозначив  $y_k = F(x_k)$  и  $y_{k-1} = F(x_{k-1})$ , запишем уравнение прямой, проходящей через две точки с координатами  $(x_k, y_k)$  и  $(x_{k-1}, y_{k-1})$

$$\frac{y - y_{k-1}}{y_k - y_{k-1}} = \frac{x - x_{k-1}}{x_k - x_{k-1}}.$$

Найдем пересечение этой прямой с осью  $Ox$ , положив  $y = 0$ . После простых преобразований найдем абсциссу пересечения этой прямой с осью  $Ox$

$$x_{k+1} = x_{k-1} - y_{k-1} \frac{x_k - x_{k-1}}{y_k - y_{k-1}}. \quad (6.27)$$

Применение метода (6.27) зависит от того, как выбраны точки  $x_k$  и  $x_{k-1}$ . Если начальные точки  $x_0, x_1$  выбраны по разные стороны от корня  $x^*$ , например,  $x_0 = a, x_1 = b$ , то мы получаем всегда сходящийся метод хорд. Если же  $x_0, x_1$  берутся по одну сторону от корня  $x^*$ , то мы получаем метод, называемый методом секущих. Достаточные условия сходимости этого метода такие же, как у метода Ньютона: требуется выполнение условий Фурье (6.21), (6.22). В этом случае сходимость метода будет монотонной. Если условия Фурье не выполняются, то метод может расходиться. Показано, что для метода секущих

$$|x_{k+1} - x^*| \leq A |x_k - x^*|^{1,618}$$

и, следовательно, его порядок сходимости  $n = 1,618$ . В отличие от метода Ньютона, в нем не требуется вычисления производных, однако для начала работы метода требуется **знать две точки  $x_0$  и  $x_1$** .

### 6.1.7. Сравнительная оценка методов

Все рассмотренные методы можно условно разделить на две группы:

1) методы, сходимость которых гарантирована в том случае, если на исходном отрезке  $[a, b]$  имеется корень, и никаких специальных условий на вид функции  $F(x)$ , кроме ее непрерывности, не накладываются;

2) методы, в которых сходимость существует только при некоторых ограничениях на вид функции  $F(x)$  и ее производные  $F'(x)$ ,  $F''(x)$ .

К первой группе из рассмотренных методов относятся метод половинного деления и метод хорд, а ко второй – метод простой итерации, метод Ньютона и метод секущих. Главное отличие этих групп методов проявляется в том, что первые – надежны, но сходятся сравнительно медленно, а вторые зависят от начального приближения и вида функции  $F(x)$ , но сходятся очень быстро. Поэтому методы второй группы, например, метод Ньютона, удобно употреблять в совокупности с другим методом, сходимость которого (хотя и сравнительно медленная) гарантирована на исходном отрезке  $[a, b]$ . Таким методом может быть метод половинного деления. В этом случае с помощью нескольких шагов метода половинного деления можно получить интервал  $[a^*, b^*]$ , такой, что при начальном приближении  $x_0$ , принадлежащим этому интервалу, метод Ньютона сходится. После этого итерации метода Ньютона позволяют получить значение корня с любой необходимой точностью. Такие гибридные методы используются во многих стандартных программах нахождения корней.

Важным параметром, по которому можно сравнивать различные методы, является общее количество работы, необходимое в каждом методе. Например, в методе половинного деления на каждой итерации требуется только вычисление с середины отрезка значения  $F(c)$  и вычисление границ нового отрезка. В методе Ньютона помимо значения  $F(x)$  требуется вычислять и значение  $F'(x)$ . Ясно, что если  $F(x)$  и  $F''(x)$  вычисляются одинаково просто, то можно использовать метод Ньютона. Однако в случае, когда  $F(x)$  вычисляется просто, а вычисление  $F'(x)$  достаточно громоздко, то имеет смысл использовать метод секущих, в котором не требуется вычислять  $F'(x)$  и который сходится так же быстро и при тех же условиях, что и метод Ньютона.

В некоторых случаях очень сложно вычисляется сама функция  $F(x)$ , для получения которой необходимо выполнение очень большого количества машинных операций. В этом случае желательно получать очень быстро сходящиеся итерационные методы, использующие как можно больше уже полученной информации. Речь идет о том, чтобы, получив, например, результаты  $x_1, x_2, \dots, x_k$  нескольких итерационных шагов, использовать их для более точного приближения корня. Кратко изложим один из наиболее известных таких методов, называемый **методом Стеффенсена** (метод, позволяющий значительно ускорить сходимость итерационных методов с помощью специально разработанной схемы расчетов).

Пусть имеется уравнение  $x = \varphi(x)$ , которое мы решаем методом итераций. Задав  $x_0$ , определим  $x_1 = \varphi(x_0)$  и  $x_2 = \varphi(x_1)$ . Стеффенсен предложил, используя три величины  $x_0, x_1, x_2$  построить специальным образом новое приближение  $y_1$ , которое будет гораздо ближе к значению корня, чем  $x_2$ . Для этого к уравнению

$$F(x) \equiv x - \varphi(x) = 0 \quad (6.28)$$

применяют метод секущих, взяв за исходные точки  $x_0$  и  $x_1$ . В результате получится новое приближение

$$y_1 = \frac{x_0 F(x_1) - x_1 F(x_0)}{F(x_1) - F(x_0)}. \quad (6.29)$$

Так как  $F(x_0) = x_0 - \varphi(x_0) = x_0 - x_1$ , а  $F(x_1) = x_1 - \varphi(x_1) = x_1 - x_2$ , то (6.29) можно записать в виде

$$y_1 = \frac{x_0 x_2 - x_1^2}{x_0 - 2x_1 + x_2}. \quad (6.30)$$

Взяв теперь  $y_1$  в качестве нового начального приближения и вычислив снова  $x_1$  и  $x_2$ , мы можем получить следующее приближение по методу Стеффенсена. Показано, что этот метод является фактически методом итераций  $y = \psi(y)$ , где  $\psi(y)$  имеет следующий вид:

$$\psi(y) = \frac{y \cdot \varphi[\varphi(y)] - \varphi(y)}{y - 2\varphi(y) + \varphi[\varphi(y)]}, \quad (6.31)$$

где надо положить  $y_0 = x_0$ , а  $y_1$  определяется по формуле (6.30).

Еще одной важной характеристикой любого метода является возможность его обобщения на случай системы многих нелинейных урав-





## Метод простой итерации

Запишем систему (6.32) в виде, удобном для итераций

$$\begin{cases} x_1 = \varphi_1(x_1, x_2, \dots, x_n), \\ x_2 = \varphi_2(x_1, x_2, \dots, x_n), \\ \dots\dots\dots \\ x_n = \varphi_n(x_1, x_2, \dots, x_n). \end{cases} \quad (6.33)$$

Решение методом простой итерации заключается в следующем. Зададимся начальным приближением  $x_1^{(0)}, x_2^{(0)}, \dots, x_n^{(0)}$ . Первое приближение вычисляется по формулам

$$\begin{cases} x_1^{(1)} = \varphi_1(x_1^{(0)}, x_2^{(0)}, \dots, x_n^{(0)}), \\ x_2^{(1)} = \varphi_2(x_1^{(0)}, x_2^{(0)}, \dots, x_n^{(0)}), \\ \dots\dots\dots \\ x_n^{(1)} = \varphi_n(x_1^{(0)}, x_2^{(0)}, \dots, x_n^{(0)}). \end{cases}$$

Затем полученное первое приближение подставляем в правую часть (6.33) и вычисляем второе приближение. В общем случае формулы метода простой итерации можно записать

$$\begin{cases} x_1^{(k+1)} = \varphi_1(x_1^{(k)}, x_2^{(k)}, \dots, x_n^{(k)}), \\ x_2^{(k+1)} = \varphi_2(x_1^{(k)}, x_2^{(k)}, \dots, x_n^{(k)}), \\ \dots\dots\dots \\ x_n^{(k+1)} = \varphi_n(x_1^{(k)}, x_2^{(k)}, \dots, x_n^{(k)}). \end{cases} \quad (6.34)$$

Итерационный процесс продолжается до тех пор, пока изменение всех неизвестных на двух последовательных итерациях не станут малыми

$$|x_i^{(k+1)} - x_i^{(k)}| < \varepsilon, (i = 1, 2, \dots, n). \quad (6.35)$$

Одно из достаточных условий сходимости метода простой итерации формулируется следующим образом: если матрица Якоби в точке  $x^*$ , которая дает решение системы (6.33), и в некоторой ее окрестности имеет все собственные значения, по модулю меньшие единицы  $|\lambda_i| < 1, (i = 1, 2, \dots, n)$ , то метод простой итерации сходится. Для выполнения этого условия достаточно, например, выполнения следующих неравенств для элементов матрицы Якоби

$$\left\{ \begin{array}{l} \left| \frac{\partial \varphi_1}{\partial x_1} \right| + \left| \frac{\partial \varphi_1}{\partial x_2} \right| + \dots + \left| \frac{\partial \varphi_1}{\partial x_n} \right| \leq q, \\ \left| \frac{\partial \varphi_2}{\partial x_1} \right| + \left| \frac{\partial \varphi_2}{\partial x_2} \right| + \dots + \left| \frac{\partial \varphi_2}{\partial x_n} \right| \leq q, \\ \dots\dots\dots \\ \left| \frac{\partial \varphi_n}{\partial x_1} \right| + \left| \frac{\partial \varphi_n}{\partial x_2} \right| + \dots + \left| \frac{\partial \varphi_n}{\partial x_n} \right| \leq q. \end{array} \right. \quad (6.36)$$

где  $q$  – некоторое положительное число, меньшее единицы, т.е.  $0 < q < 1$ . Неравенства (6.35) должны выполняться в точке  $x^*$  и в некоторой ее окрестности.

Приведем пример использования рассмотренного метода.

**Пример 6.6.** С точностью  $\varepsilon = 0,01$  решить методом итераций систему двух нелинейных уравнений с двумя неизвестными

$$\left\{ \begin{array}{l} y \cdot (x-1) = 1 \\ x^2 = y^2 + 1 \end{array} \right. \quad (6.37)$$

*Решение.* Оба уравнения (6.12) представляют собой гиперболы, вид которых приведен на Рис. 6.5. Из рисунка видно, что существует два вещественных корня. Один из них расположен в первой четверти ( $x_1 > 0, y_1 > 0$ ), а другой – в третьей ( $x_2 < 0, y_2 < 0$ ).

Для нахождения первого корня запишем систему (6.36) в виде, удобном для итераций

$$\left\{ \begin{array}{l} x = 1 + \frac{1}{y}, \\ y = \sqrt{x^2 - 1}. \end{array} \right. \quad (6.38)$$

Для численного нахождения корня  $(x_1, y_1)$  применим итерационный процесс:

$$\left\{ \begin{array}{l} x^{(k+1)} = 1 + \frac{1}{y^{(k)}} \\ y^{(k+1)} = \sqrt{(x^{(k+1)})^2 - 1} \end{array} \right. \quad (6.39)$$

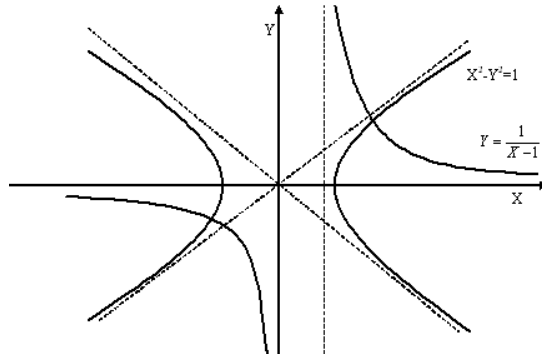


Рис. 6.5. Вид уравнений (6.36)

Из рис. 6.5 видно, что  $x_1 > 0$ ,  $y_1 > 0$ . Возьмем  $x^{(0)} = 2, y^{(0)} = \sqrt{(x^{(0)})^2 - 1} = 1,7321$ . Результаты, получаемые с помощью итерационного метода, приведены в табл. 6.4.

Таблица 6.4

#### Результаты, получаемые с помощью итерационного метода

$k$	0	1	2	3	5	...	18
$x^{(k)}$	2,0	1,5773	1,8198	1,6577	1,6926	...	1,7167
$y^{(k)}$	1,7321	1,2198	1,5204	1,3221	1,3656	...	1,3954

Видно, что итерационный процесс (6.38) сходится. Сравнивая эти результаты с приведенными ниже результатами метода Ньютона для решения этой же задачи, видно, что метод простой итерации сходится довольно медленно. Примерно одинаковых по точности результатов можно получить за 18 шагов метода простой итерации и всего за три шага метода Ньютона.

### 6.2.2. Метод Ньютона и его модификации

**Метод Ньютона** – метод, который сводит решение системы нелинейных уравнений к последовательному решению систем линейных уравнений для нахождения поправок к предыдущим приближениям.

Рассмотрим общий вид нелинейной системы (6.32). Зададим начальное приближение  $x_1^{(0)}, x_2^{(0)}, \dots, x_n^{(0)}$ . Будем искать решение системы в виде



$$J(x_1^{(0)}, \dots, x_n^{(0)}) = \begin{pmatrix} \left(\frac{\partial f_1}{\partial x_1}\right)_0 & \left(\frac{\partial f_1}{\partial x_2}\right)_0 & \dots & \left(\frac{\partial f_1}{\partial x_n}\right)_0 \\ \left(\frac{\partial f_2}{\partial x_1}\right)_0 & \left(\frac{\partial f_2}{\partial x_2}\right)_0 & \dots & \left(\frac{\partial f_2}{\partial x_n}\right)_0 \\ \dots & \dots & \dots & \dots \\ \left(\frac{\partial f_n}{\partial x_1}\right)_0 & \left(\frac{\partial f_n}{\partial x_2}\right)_0 & \dots & \left(\frac{\partial f_n}{\partial x_n}\right)_0 \end{pmatrix} \quad (6.42)$$

Если определитель этой матрицы не равен нулю, то из системы уравнений (6.41) можно найти неизвестные приращения  $\Delta x_i^{(0)}, (i=1, 2, \dots, n)$ . Поскольку в разложении Тейлора мы отбросили все члены, кроме линейных, соотношения (6.39) определяют не точное решение  $\bar{x}$ , а уточнение к нулевому приближению. Один шаг метода Ньютона заключается в решении системы линейных уравнений (6.16), все коэффициенты которой и правые части зависят от начального приближения  $x^{(0)}$ , и в получении уточненного решения  $x^{(1)}$  по формулам

$$x_1^{(1)} = x_1^{(0)} + \Delta x_1^{(0)}, x_2^{(1)} = x_2^{(0)} + \Delta x_2^{(0)}, \dots, x_n^{(1)} = x_n^{(0)} + \Delta x_n^{(0)}. \quad (6.43)$$

Приняв теперь полученное первое приближение за начальное и проделав те же операции, которые были сделаны на первом шаге, можно получить следующее приближение к решению. Этот итерационный процесс продолжается до тех пор, пока все получаемые приращения становятся малыми по абсолютной величине

$$\max_i |\Delta x_i^{(k)}| < \varepsilon (i=1, 2, \dots, n). \quad (6.44)$$

Систему линейных уравнений, которую необходимо решать на каждом шаге метода Ньютона, в общем виде можно записать следующим образом:

$$\begin{cases} \left(\frac{\partial f_1}{\partial x_1}\right)_k \Delta x_1^{(k)} + \left(\frac{\partial f_1}{\partial x_2}\right)_k \Delta x_2^{(k)} + \dots + \left(\frac{\partial f_1}{\partial x_n}\right)_k \Delta x_n^{(k)} = -f_1(x_1^{(k)}, x_2^{(k)}, \dots, x_n^{(k)}), \\ \left(\frac{\partial f_2}{\partial x_1}\right)_k \Delta x_1^{(k)} + \left(\frac{\partial f_2}{\partial x_2}\right)_k \Delta x_2^{(k)} + \dots + \left(\frac{\partial f_2}{\partial x_n}\right)_k \Delta x_n^{(k)} = -f_2(x_1^{(k)}, x_2^{(k)}, \dots, x_n^{(k)}), \\ \dots \\ \left(\frac{\partial f_n}{\partial x_1}\right)_k \Delta x_1^{(k)} + \left(\frac{\partial f_n}{\partial x_2}\right)_k \Delta x_2^{(k)} + \dots + \left(\frac{\partial f_n}{\partial x_n}\right)_k \Delta x_n^{(k)} = -f_n(x_1^{(k)}, x_2^{(k)}, \dots, x_n^{(k)}). \end{cases} \quad (6.45)$$

Индекс  $k$  у частных производных указывает, что соответствующая частная производная вычисляется при  $x_1^{(k)}, x_2^{(k)}, \dots, x_n^{(k)}$  (для значений искомых переменных на  $k$ -й итерации). Решив эту линейную систему и определив из нее величины поправок  $\Delta x_i^{(k)}, (i=1, 2, \dots, n)$ , находим следующее приближение:

$$x_i^{(k+1)} = x_i^{(k)} + \Delta x_i^{(k)}. \quad (6.46)$$

Выражения (6.45) и (6.46) полностью определяют один шаг метода Ньютона. Таким образом, метод Ньютона сводит решение систем нелинейных уравнений к многократному решению систем линейных уравнений (6.45). Можно показать, что если метод Ньютона сходится, то сходимость имеет второй порядок, как и в случае уравнения с одной неизвестной. Для сходимости метода Ньютона требуется, чтобы начальное приближение было достаточно близко к точному решению, а функции  $f_i(x_1, x_2, \dots, x_n)$  и их первые и вторые производные удовлетворяли некоторым ограничениям.

Несмотря на быструю сходимость, метод Ньютона имеет существенный недостаток: на каждом его шаге требуется вычислять  $n^2$  частных производных  $\frac{\partial f_i}{\partial x_j}$ , определяющих коэффициенты при неиз-

вестных в линейной системе (6.45). В то же время в методе простой итерации (6.34) требуется вычислять всего  $n$  функций  $\varphi_i(x_1, x_2, \dots, x_n)$ , ( $i = 1, 2, \dots, n$ ). Чтобы компенсировать этот недостаток метода Ньютона, были предложены различные его упрощения. Идея большинства из этих упрощений заключается в том, чтобы матрицу коэффициентов линейной системы вычислять не на каждом итерационном шаге, а лишь на некоторых из них, считая на остальных шагах все элементы якобиана постоянными, равными их последним вычисленным значениям на том шаге, на котором производился пересчет всех членов. В простейшем случае якобиан вычисляется один раз, в начальной точке  $x^{(0)}$ , а затем якобиан остается постоянным в течение всего итерационного процесса.

Такой метод обычно называют упрощенным методом Ньютона. Матрица коэффициентов линейной системы в этом случае остается постоянной, и в процессе итераций изменяются только правые части системы, которые пересчитываются на каждом шаге итераций и решения системы  $\Delta x_i^{(k)}$ , после нахождения которых происходит пересчет  $x_i^{(k+1)}$  по формулам (6.46). Конечно, этот метод сходится медленнее, чем метод Ньютона, но на каждом итерационном шаге требует гораздо меньше арифметических операций.

**Пример 6.7.** Решить методом Ньютона систему

$$\begin{cases} f_1 \equiv x_2(x_1 - 1) = 0, \\ f_2 \equiv x_1^2 - x_2^2 - 1 = 0, \end{cases} \quad (6.47)$$

*Решение.* Система (6.47) была решена в предыдущем примере методом итерации. Геометрическое изображение системы приведено на рис. 6.5. Система имеет два решения: в первой четверти ( $x_1 > 0, x_2 > 0$ ) и в третьей четверти ( $x_1 < 0, x_2 < 0$ ). Найдем методом Ньютона положительное решение. В качестве начального приближения примем  $x_1^{(0)} = 2, x_2^{(0)} = 1$ . Вычислим частные производные, входящие в якобиан  $\frac{\partial f_1}{\partial x_1} = x_2; \frac{\partial f_1}{\partial x_2} = x_1 - 1; \frac{\partial f_2}{\partial x_1} = 2x_1; \frac{\partial f_2}{\partial x_2} = -2x_2$ . В данном конкретном случае система (6.45) имеет вид

$$\begin{cases} (x_2)_k \Delta x_1^{(k)} + (x_1 - 1)_k \Delta x_2^{(k)} = 1 - x_2^{(k)}(x_2^{(k)} - 1), \\ (2x_1)_k \Delta x_1^{(k)} + (-2x_2)_k \Delta x_2^{(k)} = 1 - (x_1^{(k)})^2 + (x_2^{(k)})^2. \end{cases} \quad (6.48)$$

Подставляя в коэффициенты системы и правые части начальное приближение  $x_1^{(0)} = 2, x_2^{(0)} = 1$ , получим систему линейных уравнений

$$\begin{cases} 1 \cdot \Delta x_1^{(0)} + 1 \cdot \Delta x_2^{(0)} = 0, \\ 4 \cdot \Delta x_1^{(0)} - 2 \Delta x_2^{(0)} = 2. \end{cases}$$

Решив систему, получим

$\Delta x_1^{(0)} = -0,3333, \Delta x_2^{(0)} = 0,3333$ . Используя (6.46), находим уточненные значения неизвестных

$$\begin{cases} x_1^{(1)} = x_1^{(0)} + \Delta x_1^{(0)} = 2 - 0,3333 = 1,6667, \\ x_2^{(1)} = x_2^{(0)} + \Delta x_2^{(0)} = 1 + 0,3333 = 1,3333. \end{cases}$$

На этом первый шаг метода Ньютона заканчивается. Подставляя полученные значения  $x_1^{(1)}$  и  $x_2^{(1)}$  в коэффициенты и правые части системы (6.48), получим систему линейных уравнений для определения величин  $\Delta x_1^{(1)}$  и  $\Delta x_2^{(1)}$

$$\begin{cases} 1,3333x_1^{(1)} + 0,6667\Delta x_2^{(1)} = 0,1111, \\ 3,3334\Delta x_1^{(1)} - 2,6666\Delta x_2^{(1)} = 0,0000. \end{cases}$$

Решив систему получим  $\Delta x_1^{(1)} = 0,0513$  и  $\Delta x_2^{(1)} = 0,0641$ . По формулам (6.46) найдем уточненные значения  $x_1^{(1)}$  и  $x_2^{(1)}$

$$\begin{cases} x_1^{(2)} = x_1^{(1)} + \Delta x_1^{(1)} = 1,6667 + 0,0513 = 1,718, \\ x_2^{(1)} = x_2^{(1)} + \Delta x_2^{(1)} = 1,3333 + 0,0641 = 1,3974. \end{cases}$$

Представим результаты в виде табл. 6.5, причем для сравнения приведем значения решения после трех шагов Ньютона.

Рассмотрим теперь эту же задачу с точки зрения использования упрощенного метода Ньютона. Для этого матрицу коэффициентов в (6.48) вычислим один раз при  $x_1^{(0)}$  и  $x_2^{(0)}$ . Получим систему уравнений в упрощенном виде

$$\begin{cases} 1 \cdot \Delta x_1^{(k)} + 1 \cdot \Delta x_2^{(k)} = 1 - x_2^{(k)} (x_1^{(k)} - 1), \\ 4 \cdot \Delta x_1^{(k)} - 2 \Delta x_2^{(k)} = 1 - (x_1^{(k)})^2 + (x_2^{(k)})^2. \end{cases} \quad (6.49)$$

Первый шаг упрощенного метода Ньютона будет полностью совпадать с первым шагом обычного метода Ньютона, расчеты для которого проведены выше. Получим

$$\begin{cases} x_1^{(1)} = 1,6667, \\ x_2^{(1)} = 1,3333. \end{cases} \quad (6.50)$$

Сделаем теперь второй шаг упрощенного метода. Для этого подставим полученные значения (6.50) в правые части системы (6.49). Система примет вид

$$\begin{cases} 1 \cdot \Delta x_1^{(k)} + 1 \cdot \Delta x_2^{(k)} = 1 - 1,3333(1,6667 - 1) = 0,1111, \\ 4 \cdot \Delta x_1^{(k)} - 2 \Delta x_2^{(k)} = 0. \end{cases}$$

Решив эту систему получим  $\Delta x_1^{(1)} = 0,0370$  и  $\Delta x_2^{(1)} = 0,07407$ . Из (6.46) находим уточненное решение

$$\begin{cases} x_1^{(2)} = x_1^{(1)} + \Delta x_1^{(1)} = 1,6667 + 0,0370 = 1,7037, \\ x_2^{(2)} = x_2^{(1)} + \Delta x_2^{(1)} = 1,3333 + 0,07407 = 1,4074. \end{cases}$$

Без выкладок приведем результаты следующего шага упрощенного метода Ньютона:  $\Delta x_1^{(3)} = 1,7199$  и  $\Delta x_2^{(3)} = 1,4008$ . В табл. 6.5 приведены результаты счета, полученные с использованием классического и упрощенного метода Ньютона, причем для сравнения приведем значения решения упрощенным методом Ньютона на шестом шаге. Видно, что упрощенный метод в данном случае сходится примерно в 2 раза медленнее, но зато не требует на каждом шаге решения линейной системы (6.48). Для этого достаточно один раз найти обратную матрицу  $A^{-1}$  для данной системы и находить ее решение по формуле

$$x^{(k)} = A^{-1}b^{(k)},$$

где  $b^{(k)}$  – это вектор правых частей системы на  $k$  шаге.



Таблица 6.5.

**Результаты счета, полученные с использованием  
классического и упрощенного метода Ньютона**

$k$	0	1	2	3	...	6	
$x_1^{(k)}$	2	1,6667	1,718	1,7167			Классический метод Ньютона
$x_1^{(k)}$	1	1,3333	1,3974	1,3953			
$ f_1 $	0	0,1111	0,0033	0,0001			
$ f_2 $	2	0,0002	0,0012	0,0002			
$x_1^{(k)}$	2	1,6667	1,7037	...	...	1,7166	Упрощенный метод Ньютона
$x_2^{(k)}$	1	1,3333	1,4074	...	...	1,3954	
$ f_1 $	0	0,1111	0,0093	...	...	$\frac{0,0000}{6}$	
$ f_2 $	2	0,0002	0,0793	...	...	$\frac{0,0000}{4}$	

## 7. ОБЫКНОВЕННЫЕ ДИФФЕРЕНЦИАЛЬНЫЕ УРАВНЕНИЯ

### 7.1. Общие понятия

Обыкновенным дифференциальным уравнением называют такое уравнение, которое содержит аргумент  $x$ , искомую функцию  $y(x)$  и одну или несколько производных от искомой функции:

$$F(x, y, y', \dots, y^{(n)}) = 0. \quad (7.1)$$

Наивысший порядок  $n$  входящей в уравнение производной называется порядком дифференциального уравнения. Если удастся в явном виде выразить старшую производную, то такое уравнение называется уравнением, разрешенным относительно старшей производной

$$y^{(n)} = \psi(x, y, y', \dots, y^{(n-1)}). \quad (7.2)$$

Решением дифференциального уравнения (7.1) называется такая функция  $y(x)$ , которая после подстановки ее в уравнение превращает его в тождество. Важно отметить, что в общем случае дифференциальное уравнение имеет бесчисленное множество решений. Общее решение уравнения (7.1) содержит  $n$  произвольных постоянных и имеет вид

$$y = \varphi(x, C_1, C_2, \dots, C_n). \quad (7.3)$$

Каждое частное решение дифференциального уравнения получается из общего, если произвольные постоянные принимают определенные значения.

Для выделения единственного решения дифференциального уравнения, т.е. для получения частного решения, необходимо задать  $n$  произвольных постоянных. При аналитическом решении дифференциальных уравнений методами высшей математики определяется общее решение, зависящее от  $n$  произвольных постоянных, после чего они находятся с помощью  $n$  дополнительных условий. К сожалению, получить общее решение в аналитическом виде можно лишь в небольшом количестве случаев: для линейных уравнений с постоянными коэффициентами, для некоторых классов уравнений с переменными коэффициентами (как линейных, так и нелинейных). В большинстве случаев, встречающихся на практике, дифференциальные уравнения приходится решать численно. В этом случае, используя уравнение и дополнительные условия, сразу строится частное решение задачи.

В зависимости от вида дополнительных условий для получения частного решения дифференциального уравнения существуют два различных типа задач: **задача с начальными данными (задача Коши)** и **краевая задача**. В качестве дополнительных условий обычно задаются значения искомой функции и ее производных при некоторых значениях независимой переменной. Если все эти дополнительные условия задаются в одной точке, то такая задача называется задачей Коши, а сами эти условия называются начальными условиями. Если же дополнительные условия задаются более чем в одной точке, то такая задача называется краевой, а сами дополнительные условия называются краевыми.

выми (или граничными) условиями. Такое название принято потому, что в большинстве случаев краевые условия задаются в точках, являющихся границами области решения дифференциального уравнения.

Примеры задач Коши:

$$\frac{dy}{dx} = x - \sin^2(xy), \quad x > 0, y(0) = 2; \quad (7.4)$$

$$y'' + 2 \cos xy' - xy = x^2, \quad x > 1, y(1) = 4, y'(1) = 1. \quad (7.5)$$

В первой задаче уравнение имеет первый порядок и поэтому задано одно дополнительное условие. Следующее дифференциальное уравнение имеет второй порядок и поэтому заданы два дополнительных условия в задаче Коши, причем оба в точке  $x = 1$ .

Примеры краевых задач:

$$y'' + e^x y' - 3y = \cos x, \quad x \in [0,1], y(0) = 3, y(1) = 2; \quad (7.6)$$

$$y^{(4)} - 2 \sin xy'' + y' = \operatorname{tg}^2 x, x \in [a, b], \quad (7.7)$$

$$y(a) = y'(a) = 0, y(b) = y'(b) = 2$$

Первое уравнение имеет второй порядок и для него заданы два дополнительных условия в граничных точках  $x = 0$  и  $x = 1$ . Второе уравнение имеет четвертый порядок и для него заданы четыре граничных условия в точках  $x = a$  и  $x = b$ .

Среди численных методов решения дифференциальных уравнений чаще всего встречаются **метод конечных разностей** и **метод конечных элементов**. В обоих случаях решение задачи ищется в виде дискретной функции, определенной на дискретном множестве аргументов, и сводится, в конечном счете, к решению систем линейных алгебраических уравнений (для линейных дифференциальных уравнений) или систем нелинейных уравнений (для нелинейных дифференциальных уравнений). Отличие обоих методов заключается в способе получения систем уравнений. В методе конечных разностей входящие в дифференциальное уравнение производные аппроксимируются конечными разностями, и задача сводится к системе разностных уравнений. В методе конечных элементов краевая задача заменяется некоторой эквивалентной ей задачей, в которой требуется найти минимум функционала для функции, определенной на дискретном множестве точек. Мы будем в основном рассматривать конечно-разностные методы. Рассмотрим сущность этих методов и основные понятия, которые используются при их применении.

## 7.2. Метод конечных разностей

Сущность метода состоит в следующем. Область непрерывного изменения аргумента (например, отрезок  $[a, b]$ , на котором определено решение  $y(x)$  дифференциального уравнения) заменяется дискретным множеством точек, называемых узлами. Эти узлы образуют разностную сетку. Искомая функция  $y(x)$ , являющаяся решением дифференциального уравнения и непрерывно зависящая от  $x$ , заменяется функцией дискретного аргумента на данной сетке. Эта функция называется **сеточной**. Исходное дифференциальное уравнение заменяется разностным уравнением относительно сеточной функции путем замены входящих в уравнение производных соответствующими конечно-разностными формулами. Замена дифференциального уравнения разностным называется его **разностной аппроксимацией**. Таким образом, решение дифференциального уравнения сводится к решению разностных уравнений относительно значений сеточной функции в узлах сетки. Для возможности практического использования метода конечных разностей необходимо:

- обосновать возможность замены дифференциального уравнения разностным;
- оценить точность получаемых решений;
- исследовать устойчивость метода.

В итоге все перечисленные исследования и оценки должны помочь сформулировать условия, при выполнении которых решение разностных уравнений сходится к решению дифференциального уравнения с соответствующими граничными или начальными условиями при неограниченном уменьшении шага сетки.

Для удобного и компактного описания используют операторную запись дифференциальных и разностных уравнений и соответствующих начальных и граничных условий, которая является символическим видом их записи. Приведенные выше в качестве примера дифференциальные уравнения (7.4)–(7.7) можно записать в символическом виде

$$Ly = \varphi(x), x \in G. \quad (7.8)$$

Здесь  $L$  – дифференциальный оператор, зависящий от функции  $y(x)$  и содержащий функции дифференцирования. Его значение различно для различных дифференциальных уравнений. Функция  $\varphi(x)$  называется правой частью операторного уравнения, она зависит от аргумента  $x$ . Символом  $G$  обозначена область изменения аргумента. В частном случае область  $G$  – это отрезок  $[a, b]$  или полуось  $x > 0$ . Дополнительные условия на границе также представляются в операторном виде:

$$ly = \varphi(x), x \in Q. \quad (7.9)$$

Здесь  $l$  – оператор начальных или граничных условий, который зависит от функции  $y(x)$  и, быть может, некоторых ее производных,  $\varphi(x)$  – правая часть граничных условий,  $Q$  – множество, задающее границу области  $G$  (например,  $x = 0$  и  $x = 1$  или  $x = a$  и  $x = b$ ). Поясним операторную запись (7.8), (7.9) на примере краевой задачи (7.6). В этом случае:

$$Ly = y'' + e^x y' - 3y, f(x) = \cos x, G = \{0 \leq x \leq 1\},$$

$$ly = \begin{cases} y(0) \\ y(1) \end{cases} \quad \varphi(x) = \begin{cases} 3 \\ 2 \end{cases} \quad Q = \{x = 0, x = 1\}.$$

При использовании разностного метода дифференциальное уравнение (7.8) заменяется разностным уравнением, которое также можно записать в операторном виде

$$L_h y_h = f_h(x), x \in g_h \quad (7.10)$$

Здесь  $h L$  – разностный оператор, аппроксимирующий дифференциальный оператор  $L$ ;  $g_h \{x_i\} (i = 0, 1, \dots, n)$  – дискретное множество аргументов, расположенных на одинаковом расстоянии  $h$  друг от друга;  $y_h$  – дискретная сеточная функция, определенная на  $g_h$ , с помощью которой мы заменяем непрерывную функцию  $y(x)$ , являющуюся решением дифференциального уравнения (7.10);  $f_h(x_i)$  – сеточная функция, определенная на  $g_h$  и заменяющая непрерывную функцию  $f(x)$ . Индекс  $h$  в (7.10) подчеркивает, что величина шага сетки является параметром разностной задачи. Следовательно, (7.10) можно рассматривать как целое семейство разностных уравнений, которые зависят от параметра  $h$ . При рассмотрении формул численного дифференцирования отмечалось, что погрешность аппроксимации производной в некоторой точке  $x_i$  может быть представлена в виде  $O(h^k)$ . В этом случае, имея в виду весь дифференциальный оператор, говорят, что в точке  $x_i$  имеет место **аппроксимация  $k$ -го порядка**.

При численном решении дифференциальных уравнений требуется оценить погрешность аппроксимации не в одной точке, а на всей сетке, т.е. в узлах  $x_0, x_1, \dots, x_n$ . В качестве погрешности аппроксимации  $\varepsilon_h$  на всей сетке можно принять некоторую интегральную величину, связанную с погрешностями аппроксимации в отдельных узлах.

Наиболее часто в качестве такой величины принимается одна из следующих величин:

$$\varepsilon_h = \max_i |\varepsilon(x_i)|, \quad (7.11)$$

$$\varepsilon_h = \left[ \sum_{i=1}^n \varepsilon^2(x_i) \right]^{1/2}. \quad (7.12)$$

Величина (7.11) означает, что в качестве меры интегральной аппроксимации на всем отрезке принимается наибольшая из погрешностей в отдельных узлах сетки, а (7.12) означает выбор среднеквадратичной величины.

Наряду с аппроксимацией (7.10) дифференциального уравнения (7.8) вводится аппроксимация начальных или граничных условий (7.9), которые можно записать в виде

$$l_h y_h = \varphi(x), \quad x \in q_h. \quad (7.13)$$

Здесь  $q_h$  – граничные узлы сетки, а индекс  $h$  означает зависимость разностной аппроксимации граничных условий от значения шага  $h$ .

Совокупность разностных уравнений (7.10) и (7.13), аппроксимирующих исходное дифференциальное уравнение и дополнительные условия на границах, называется **разностной схемой**.

Главный вопрос, на который необходимо ответить при использовании разностных схем, заключается в том, будет ли решение разностной задачи  $h$  у сходиться к точному решению дифференциальной задачи  $y(x)$  во всех узловых точках сетки при неограниченном измельчении сетки, т.е. при  $h \rightarrow 0$ . Для ответа на этот вопрос необходимо рассмотреть такие понятия, как сходимость, устойчивость и аппроксимация разностной схемы.

Аналогично величине  $\varepsilon_h$  можно ввести величину погрешности замены краевых или начальных условий их разностными соотношениями. Далее под величиной  $\varepsilon_h$  мы будем понимать интегральную величину общей погрешности, вносимой как разностной схемой, где учитываются ошибки, вносимые заменой дифференциального уравнения разностным, так и заменой дополнительных условий (начальных или граничных) конечно-разностными выражениями.

Будем говорить, что разностная схема (7.10), (7.13) аппроксимирует дифференциальное уравнение (7.) и дополнительные условия на границах (7.9), если

$$\varepsilon_h \rightarrow 0 \text{ при } h \rightarrow 0 \quad (7.14)$$

Если при этом

$$|\varepsilon_h| \leq ch^k, \quad (7.15)$$

где  $c > 0$  и  $k > 0$  – некоторые постоянные, не зависящие от величины шага  $h$ , то говорят, что аппроксимация имеет порядок  $k$  относительно шага сетки  $h$ .

Разностная схема называется **устойчивой**, если малому изменению входных данных соответствует малое изменение решения разностных уравнений. Входными данными являются коэффициенты в разностных уравнениях, их правые части и начальные или граничные условия. В противном случае разностная схема называется **неустойчивой**.

Говорят, что решение разностной краевой задачи (7.10), (7.13) сходится к решению дифференциальной задачи (7.8), (7.9), если

$$\max_i |y(x_i) - y_h(x_i)| \leq Mh^k. \quad (7.16)$$

Здесь  $y(x_i)$  – точное решение дифференциальной задачи в точке  $x = x_i$ , а  $y_h(x_i)$  – решение разностной задачи в этой же точке.

Основной результат, полученный в теории разностных схем, заключается в том, что если разностная схема аппроксимирует исходную дифференциальную задачу и является устойчивой, то решение, полученное по разностной схеме, сходится к решению дифференциальной задачи при неограниченном измельчении сетки, т.е. при  $h > 0$ .

Другими словами, из устойчивости и аппроксимации разностной схемы следует ее сходимости. Это означает, что трудную задачу исследования сходимости разностной схемы к решению дифференциальной задачи можно заменить более простыми задачами изучения аппроксимации схемы и исследования ее устойчивости.

### 7.3. Решение задачи Коши

Рассмотрим дифференциальное уравнение первого порядка, разрешенного относительно производной

$$\frac{dy}{dx} = f(x, y), \quad (7.17)$$

удовлетворяющего начальному условию (задача Коши)

$$y(x_0) = y_0. \quad (7.18)$$

Решением задачи Коши является функция  $y(x)$ , которая удовлетворяет уравнению (7.18) и начальному условию (7.19). Геометрически это означает, что из множества функций, удовлетворяющих уравнению (7.18), мы находим такое частное решение, которое проходит через точку с координатами  $(x_0, y_0)$  (рис. 7.1). Используем для решения сформулированной задачи конечно-разностный метод. Введем узловые точки  $x_0, x_1, \dots, x_n$ , отстоящие друг от друга на величину постоян-

ного шага  $h$ . Рассмотрим сеточную функцию  $y_0, y_1, \dots, y_n$ , определенную в узлах сетки. Используя различные разностные аппроксимации для производной  $dy/dx$ , мы можем получить различные разностные уравнения. Их общий вид

$$y_{i+1} = \varphi(x_i, h, y_{i+1}, y_i, y_{i-1}, \dots, y_{i-k+1}). \quad (7.19)$$

Конкретный вид и тип разностной схемы зависят от того, какие из выписанных аргументов в правой части уравнения в ней в действительности присутствуют. Если в правой части (7.20) отсутствует величина  $y_{i+1}$ , то схема называется явной. Это означает, что значение  $y_{i+1}$ , стоящее в левой части, явно вычисляется по предыдущим значениям  $y_i, y_{i-1}, \dots, y_{i-k+1}$ .

В случае, когда  $k = 1$ , мы получаем **одношаговый метод**

$$y_{i+1} = \varphi(x_i, h, y_i). \quad (7.20)$$

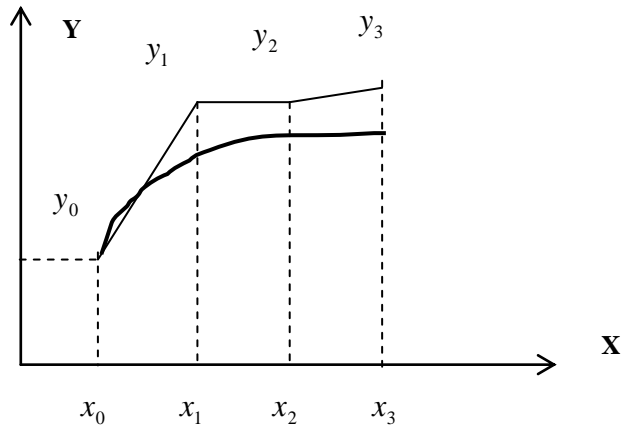


Рис. 7.1. Частное решение множества функций, удовлетворяющих уравнению (7.18)

В одношаговом методе решение в очередной точке вычисляется с использованием значения функции в предыдущей точке.

При  $k = 2$  получаем двухшаговый метод

$$y_{i+1} = \varphi(x_i, h, y_i, y_{i-1}). \quad (7.21)$$

В одношаговом методе значение начального условия  $y(x_0) = y_0$  позволяет последовательно определить значения сеточной функции во



всех последовательных узлах по формуле (7.21):  $y_1 = \varphi(x_0, h, y_0)$ ,  $y_2 = \varphi(x_1, h, y_1)$ ,  $y_3 = \varphi(x_2, h, y_2)$ , ...,  $y_n = \varphi(x_{n-1}, h, y_{n-1})$ . В двухшаговом методе (7.22) для вычисления значения  $y_{i+1}$  требуется знание двух предыдущих значений функции  $y_i, y_{i-1}$ , и поэтому одного начального значения  $y_0$  для определения следующего значения  $y_i$  здесь уже недостаточно. Аналогично обстоит дело и для  $k$ -шаговых методов при  $k = 3$ .

В **многошаговом методе решения задачи Коши** для вычисления очередного решения используется несколько предыдущих уже полученных значений решения в нескольких точках.

Если в правую часть (7.20) входит искомое значение  $y_{i+1}$ , то метод называется неявным. В этом случае (7.20) не позволяет явно вычислить величину  $y_{i+1}$  и для ее нахождения приходится использовать итерационные методы.

#### 7.4. Метод Эйлера

Рассмотрим наиболее простой явный одношаговый метод, который называют **методом Эйлера**. Будем решать уравнение (7.18) с начальным условием (7.19) на введенной ранее сетке с постоянным шагом  $h$ . Запишем в начальной точке  $x_0$  значение первой производной с помощью правой разности  $y'(x_0) = \frac{y_1 - y_0}{h} + O(h)$ . Используя это выражение, запишем разностное уравнение, приближенно аппроксимирующее уравнение (7.18) в точке  $x_0$

$$\frac{y_1 - y_0}{h} + O(h) = f(x_0, y_0). \quad (7.22)$$

Выражая из этого уравнения неизвестную величину  $y_1$ , получим

$$y_1 = y_0 + hf(x_0, y_0) + O(h^2). \quad (7.23)$$

Поскольку  $x_0$  и  $y_0$  известны из начальных условий, то, пренебрегая последним членом, определяющим погрешность аппроксимации дифференциального уравнения разностным, получим формулу метода Эйлера

$$y_1 = y_0 + hf(x_0, y_0). \quad (7.24)$$

Так как  $x_1 = x_0 + h$ , то (7.25) определяет следующую точку  $(x_1, y_1)$ , которую мы можем принять за начальную для следующего шага  $y_2 = y_1 + hf(x_1, y_1)$ . Аналогично может быть найдено значение сеточной функции в любом узле сетки

$$y_{i+1} = y_i + hf(x_i, y_i). \quad (7.25)$$

Как следует из (7.24), метод Эйлера имеет на каждом шаге локальную ошибку  $O(h^2)$ . Для определения величины  $y_n$  нам придется сделать  $n$  шагов по формуле (7.26), на каждом из которых погрешность имеет порядок  $O(h^2)$ , и, таким образом, за  $n$  шагов погрешность будет вычисляться выражением

$$R = nO(h^2) \approx \frac{1}{h} O(h^2) \approx O(h), \text{ т.е. глобальная погрешность за } n \text{ шагов}$$

имеет погрешность первого порядка  $O(h)$ . Согласно формуле для погрешности, для ее уменьшения надо уменьшать шаг  $h$ , однако в этом случае увеличивается общее количество шагов и могут накапливаться погрешности округления.

Метод Эйлера имеет простую геометрическую интерпретацию. Так как  $f(x_0, y_0) = y'_0$ , то из геометрического смысла производной следует, что  $f(x_0, y_0) = \operatorname{tg} \alpha$ , где  $\alpha$  – угол наклона касательной к кривой  $y(x)$ , являющейся решением дифференциального уравнения в точке  $(x_0, y_0)$ . Поэтому результат первого шага  $y_1$  в методе Эйлера (формула (7.25)) есть пересечение касательной к решению дифференциального уравнения в точке  $x_0$  с прямой  $x = x_1$ . Следовательно, метод Эйлера заменяет кривую  $y(x)$  на отрезке  $[x_0, x_1]$  прямой линией, являющейся касательной к функции  $y(x)$  в точке  $(x_0, y_0)$ , а истинное решение  $y(x)$  на отрезке  $[x_0, x_n]$  ломаной (рис.7.1), которая с ростом  $x$  может все дальше уходить от решения  $y(x)$ . Отсюда же видно, как можно усовершенствовать метод Эйлера. Для этого надо учесть изменение угла наклона касательной к кривой  $y(x)$  при перемещении от точки  $x_0$  к точке  $x_1$ , а не брать этот угол постоянным, равным  $\operatorname{arctg}(f(x_0, y_0))$ , как это сделано в методе Эйлера.

## 7.5. Метод Эйлера с пересчетом

Одним из возможных улучшений метода Эйлера является **метод Эйлера с пересчетом**, который для получения величины  $y_1$  использу-

ет два этапа. На первом этапе определяется предварительное значение  $\tilde{y}_1$  по методу Эйлера

$$\tilde{y}_1 = y_0 + hf(x_0, y_0). \quad (7.26)$$

Имея теперь точки  $(x_0, y_0)$  и  $(x_1, \tilde{y}_1)$ , мы определим значения производной  $y'(x) = f(x, y)$  в этих точках, которые будут равны  $f(x_0, y_0)$  и  $f(x_1, \tilde{y}_1)$ , и возьмем среднее значение этих производных  $f_c = \frac{1}{2}(f(x_0, y_0) + f(x_1, \tilde{y}_1))$ . На втором этапе определяется уточненное значение  $y_1$  по формуле

$$y_1 = y_0 + hf_c = y_0 + \frac{h}{2}(f(x_0, y_0) + f(x_1, \tilde{y}_1)). \quad (7.27)$$

Геометрически формула (7.28) означает, что мы опять получаем решение на отрезке  $[x_0, x_1]$  в виде прямой, но теперь наклон этой прямой определяется средним значением производных в начальной точке  $x_0$  и конечной точке  $x_1$ , а не только в начальной точке, как это сделано в методе Эйлера. В общем виде один шаг метода Эйлера с пересчетом можно записать следующим образом:

$$\tilde{y}_{i+1} = y_i + hf(x_i, y_i). \quad (7.28)$$

$$y_{i+1} = y_i + \frac{h}{2}(f(x_i, y_i) + f(x_{i+1}, \tilde{y}_{i+1})). \quad (7.29)$$

Метод Эйлера с пересчетом является частным случаем общего подхода к решению задачи Коши, называемого методом прогноза и коррекции. В данном случае определение предварительного значения  $\tilde{y}_{i+1}$  является прогнозом истинной величины  $y_{i+1} = y(x_{i+1})$ , где  $y(x)$  – точное решение задачи. Второй этап метода является корректировкой предварительного прогноза  $\tilde{y}_{i+1}$  и дает определение более точного значения  $y_{i+1}$ , которое гораздо ближе к точному решению. Можно показать, что локальная погрешность метода Эйлера с пересчетом будет иметь порядок  $O(h^3)$ , а глобальная погрешность на отрезке  $[x_0, x_n]$  – порядок  $O(h^2)$ . Этот метод является первой итерацией более общего подхода, определяемого формулой

$$y_{i+1} = y_i + \frac{h}{2}(f(x_i, y_i) + f(x_{i+1}, y_{i+1})). \quad (7.30)$$

Этот метод является неявным, так как искомая величина  $y_{i+1}$  входит как в левую, так и в правую часть уравнения. Можно построить

итерационный метод определения  $y_{i+1}$  из (7.31), но поскольку такая итерационная процедура должна применяться на каждом шаге решения задачи Коши, то метод будет достаточно громоздким.

Метод Эйлера с пересчетом реализует один шаг метода итераций. С помощью этого метода можно проводить контроль точности решения путем сравнения значений  $\tilde{y}_{i+1}$  и  $y_{i+1}$  и выбора на основании этого величины шага  $h$  в каждом узле  $x_i$ . Для этого необходимо на каждом шаге вычислять величину  $|y_{i+1} - \tilde{y}_{i+1}|$ . Если эта величина сравнима с погрешностями вычислений, то шаг  $h$  нужно увеличить; в противном случае, если эта разность слишком велика, значение  $h$  следует уменьшить. С помощью процедур такого типа можно реализовать автоматический выбор шага при решении задачи Коши.

### 7.6. Метод Рунге–Кутта

Помимо рассмотренных существуют и другие явные одношаговые методы. На практике чаще всего используется метод Рунге–Кутта, который для гладких функций обладает очень высокой локальной точностью  $O(h^5)$  и суммарной точностью на отрезке  $[x_0, x_n]$  порядка  $O(h^4)$ . Для того чтобы привести формулы этого метода, заметим, что метод Эйлера с пересчетом можно записать в следующей компактной форме:

$$y_{i+1} = y_i + \Delta y_i. \quad (7.31)$$

Тогда для метода Эйлера

$$\Delta y_i = h_i f(x_i, y_i). \quad (7.32)$$

а для метода Эйлера с пересчетом

$$\Delta y_i = \frac{h}{2} (K_1 + K_2). \quad (7.33)$$

$$K_1 = f(x_i, y_i), \quad K_2 = f(x_i + h_i, y_i + K_1 h_i). \quad (7.34)$$

В формулах (7.35) производные  $K_1$  и  $K_2$  вычисляются в начальной и конечной точках интервала  $[x_i, x_i + h_i]$ . Метод Рунге–Кутта является обобщением такого подхода. В нем производные вычисляются как в крайних точках, так и середине интервала  $[x_i, x_{i+1}]$ , а затем берется некоторое среднее значение

$$\Delta y_i = \frac{h}{6} (K_1 + 2K_2 + 2K_3 + K_4); \quad (7.35)$$

$$\begin{aligned}
 K_1 &= f(x_i, y_i), & K_2 &= f\left(x_i + \frac{h_i}{2}, y_i + \frac{K_1 h_i}{2}\right); \\
 K_3 &= f\left(x_i + \frac{h_i}{2}, y_i + \frac{K_2 h_i}{2}\right), & K_4 &= f(x_i + h_i, y_i + K_3 h_i).
 \end{aligned}
 \tag{7.36}$$

Коэффициенты при  $K_i (i=1,2,3,4)$  в (7.36) определены теоретически из условия получения наиболее высокой точности метода. Метод Рунге–Кутты требует четырехкратного вычисления производных, т.е. функции  $f(x, y)$ , по формулам (7.37). Однако это окупается его высокой точностью, что позволяет решать дифференциальное уравнение с гораздо большим шагом, чем в методах Эйлера и Эйлера с пересчетом, при сравнимой точности решений. Кроме того, метод Рунге–Кутты легко позволяет в процессе решения изменять шаг  $h_i$  в зависимости от получаемой гладкости решения: если получаемое решение сильно растёт, то шаг  $h_i$  уменьшается, если же решение изменяется слабо при переходе от  $x_i$  к  $x_{i+1} = x_i + h_i$ , то шаг  $h_i$  будет увеличиваться. В различных системах математического обеспечения ЭВМ имеются хорошие и эффективные программы, реализующие метод Рунге–Кутты с переменным шагом и заданной точностью вычислений.

Приведем пример решения задачи Коши с помощью трех рассмотренных методов и сравним точность получаемых результатов.

**Пример 7.1.** Решить задачу Коши

$$y' = 2(x^2 + y); y(0); x \in [0, 1]; h = 0,1. \tag{7.37}$$

*Решение.* Прежде всего заметим, что эту задачу можно решить известными методами из курса высшей математики. Ее точное решение с учетом начального условия имеет вид  $y = 1,5e^{2x} - x^2 - x - 0,5$ . Для проведения численных расчетов заметим, что в нашем случае  $f(x, y) = 2(x^2 + y)$ .

а. Метод Эйлера. Определяем результат первого шага

$$y_1 = y_0 + hf(x_0, y_0) = 1 + 0,1 \cdot (2 \cdot (0 + 1))$$

На втором шаге получаем

$$y_2 = y_1 + hf(x_1, y_1) = 1,2 + 0,1 \cdot (2 \cdot (0,1^2 + 1,2)) = 1,4420.$$

Результаты счета при других  $x$  приведены в табл. 7.1.

б. Метод Эйлера с пересчетом. Результаты первого шага:

$$\begin{aligned}\tilde{y}_1 &= y_0 + hf(x_1, y_1) = 1 + 0,1(2 \cdot (0+1)) = 1,2; \\ y_1 &= y_0 + \frac{h}{2}(f(x_0, y_0) + f(x_1, \tilde{y}_1)) = \\ &= 1 + \frac{0,1}{2}(2 \cdot (0+1) + 2 \cdot (0,1^2 + 1,2)) = 1,2210.\end{aligned}$$

Результаты второго шага:

$$\begin{aligned}\tilde{y}_2 &= y_1 + hf(x_2, y_2) = 1,221 + 0,1(2 \cdot (0,1^2 + 1,2210)) = 1,4672; \\ y_2 &= y_1 + \frac{h}{2}(f(x_1, y_1) + f(x_2, \tilde{y}_2)) = \\ &= 1,221 + \frac{0,1}{2}(2 \cdot (0,1^2 + 1,221) + 2 \cdot (0,2^2 + 1,4672)) = 1,4923.\end{aligned}$$

Результаты счета при других  $x$  приведены в табл. 7.1.

в. Метод Рунге–Кутта. Приведем результаты вычисления одного шага:

$$\begin{aligned}K_1 &= f(x_0, y_0) = 2 \cdot (0+1); \\ K_2 &= f(x_0 + \frac{h}{2}, y_0 + \frac{K_1 h}{2}) = 2((0 + \frac{0,1}{2})^2 + 1 + \frac{2 \cdot 0,1}{2}) = 2,205; \\ K_3 &= f(x_0 + \frac{h}{2}, y_0 + \frac{K_2 h}{2}) = 2 \cdot ((0 + \frac{0,1}{2})^2 + 1 + \frac{2,205 \cdot 0,1}{2}) = 2,2255; \\ K_4 &= f(x_0 + h, y_0 + K_3 h) = 2 \cdot (0,1^2 + 1 + 2,2255 \cdot 0,1) = 2,4651; \\ y_1 &= y_0 + \frac{h}{6}(K_1 + 2K_2 + 2K_3 + K_4) = \\ &= 1 + \frac{0,1}{6}(2 + 2 \cdot 2,205 + 2 \cdot 2,2255 + 2,4651) = 1,2221\end{aligned}$$

В табл. 7.1 приведены результаты всех трех методов на интервале  $[0, 1]$  с постоянным шагом  $h = 0,1$  и значения точного решения.

Результаты, приведенные в таблице, показывают очень высокую точность метода Рунге–Кутта и нарастание погрешностей метода Эйлера с увеличением  $x_i$ . При  $x = 1$  относительные погрешности  $|y - y_T|/y_T$ , где  $y_T$  – точное решение, равны: для метода Эйлера – 0,18; для метода Эйлера с пересчетом – 0,068; для метода Рунге–Кутта – 0,000023.

Сравнивая работу, необходимую для вычисления одного шага по каждому из методов, отметим, что для метода Эйлера следует один раз вычислить правую часть уравнения  $f(x, y)$ , для метода с пересчетом – 2

раза, а для метода Рунге–Кутта – 4 раза. Однако лишняя работа в методе Рунге–Кутта с лихвой окупается его высокой точностью, что позволяет проводить вычисления с существенно большими шагами, чем в методе Эйлера или в методе Эйлера с пересчетом.

Таблица 7.1.

**Результаты применения всех трех методов  
на интервале**

$x_i$	Метод Эйлера	Метод Эйлера с пересчетом	Метод Рунге–Кутта	Точное решение
0,1	1,2000	1,2210	1,2221	1,2221
0,2	1,4420	1,4923	1,4977	1,4977
0,3	1,7384	1,8284	1,8432	1,8432
0,4	2,1041	2,2466	2,2783	2,2783
0,5	2,5569	2,7680	2,8274	2,8274
0,6	3,1183	3,4176	3,5201	3,5202
0,7	3,8139	4,2257	4,3927	4,3928
0,8	4,6747	5,2288	5,4894	5,4895
0,9	5,7376	6,4004	6,8643	6,8645
1,0	7,0472	8,0032	8,5834	8,5836

### 7.7. Многошаговые методы

Недостатком метода Рунге–Кутта является необходимость вычислять 4 раза правую часть уравнения  $f(x, y)$ . От этого недостатка свободны **многошаговые методы**, которые для вычисления значения решения  $y_{i+1}$  используют значения решения не в одной точке  $y_i$ , как это делается в одношаговых методах, а в нескольких предыдущих точках  $y_i, y_{i-1}, y_{i-2}, \dots, y_{i-k+1}$ . Идея построения многошаговых методов заключается в следующем. Рассмотрим исходное уравнение  $y' = f(x, y)$  и проинтегрируем обе части этого уравнения по  $x$  на отрезке  $[x_i, x_{i+1}]$ . Интеграл по левой части вычисляется сразу и имеет вид  $\int_{x_i}^{x_{i+1}} y' dx = y_{i+1} - y_i$ , а интеграл по правой части – путем предварительного построения интерполяционного многочлена  $P_{k-1}(x)$  степени  $k - 1$ , который аппроксимирует функцию  $f(x, y)$  на отрезке  $[x_i, x_{i+1}]$  по значениям этой функции в нескольких предыдущих точках, решения в которых уже получены:  $f(x_{i-1}, y_{i-1}), \dots, f(x_{i-k+1}, y_{i-k+1})$ . Тогда

$y_{i+1} - y_i = \int_{x_i}^{x_{i+1}} f(x, y) dx \approx \int_{x_i}^{x_{i+1}} P_{k-1}(x) dx$ . На основе этой формулы можно

строить различные многошаговые методы любого порядка точности. Порядок точности зависит от степени интерполяционного многочлена, для построения которого используются значения сеточной функции  $y_i, y_{i-1}, y_{i-2}, \dots, y_{i-k+1}$ , вычисленные на  $k$  предыдущих шагах. Приведем без вывода две расчетные формулы из семейства явных многошаговых методов Адамса-Башфорта.

Двухшаговый метод второго порядка точности записывается в виде

$$y_{i+1} = y_i + \frac{h}{2}(3f(x_i, y_i) - f(x_{i-1}, y_{i-1})). \quad (7.38)$$

Более точный четырехшаговый метод четвертого порядка точности описывается формулой

$$y_{i+1} = y_i + \frac{h}{24}(55f(x_i, y_i) - 59f(x_{i-1}, y_{i-1}) + 37f(x_{i-2}, y_{i-2}) + 37f(x_{i-3}, y_{i-3})). \quad (7.39)$$

Для вычисления  $y_{i+1}$  по формуле (7.39) необходимо знать решения в двух предыдущих точках  $y_{i-1}$  и  $y_{i-1}$ , а по (7.40) – решения в четырех предыдущих точках. После этого можно вычислить правые части уравнения  $f(x, y)$  в этих точках  $f(x_i, y_i)$ ,  $f(x_{i-1}, y_{i-1})$ ,  $f(x_{i-2}, y_{i-2})$  и воспользоваться формулами (7.39) и (7.40) для определения  $y_{i+1}$ . Отметим, что переходя к вычислению на следующем шаге величины  $y_{i+2}$ , достаточно вычислить функцию  $f(x, y)$  только в одной точке  $f(x_{i+1}, y_{i+1})$ , так как предыдущие значения  $f(x_i, y_i)$ ,  $f(x_{i-1}, y_{i-1})$ ,  $f(x_{i-2}, y_{i-2})$  уже вычислены.

Недостаток многошаговых схем заключается в том, что невозможно начать счет по одному лишь известному значению  $y_0$ , которое задано в начальном условии. Например, в (7.40) расчет можно начинать лишь со значения  $y_4$  в точке  $x_4$ . Необходимые для такого расчета значения  $y_0, y_1, y_2$  и  $y_3$  должны быть вычислены каким-либо другим методом (например, методом Эйлера или Рунге–Кутты). При этом начальные точки надо вычислить очень точно, так как иначе полученные в начальных точках погрешности будут влиять на решение задачи во всех остальных точках. Другой недостаток многошаговых методов заключается в том, что они, как правило, не позволяют изменять шаг в



процессе расчета (без существенного усложнения алгоритма расчета), чтобы учесть характер получаемого решения, подстраивая шаг  $h$  к скорости изменения решения.

## Лабораторный практикум

### Лабораторная работа № 1

#### Решение систем линейных алгебраических уравнений точными методами

**Цель работы:** изучение точных методов решения СЛАУ и формирование навыков их реализации на ЭВМ.

#### Содержание работы

1. Изучить теоретический материал.
2. Найти вручную точное решение системы линейных уравнений, указанной в индивидуальном задании.
3. Создать исходный модуль программы на языке высокого уровня Паскаль, отладить его, просчитать тестовый пример.
4. Получить посредством разработанного модуля решение индивидуального задания и сравнить его с точным решением.
5. Оформить отчет.

#### Содержание отчета

1. Тема и цель работы, содержание индивидуального задания.
2. Этапы получения точного решения вручную.
3. Блок-схема алгоритма.
4. Исходный текст программы и результаты ее тестирования.
5. Результаты работы программы согласно индивидуальному заданию.

#### Варианты индивидуальных заданий

Найти решение системы линейных алгебраических уравнений вида  $Ax = \bar{b}$  методом Гаусса, если значения матрицы  $A$  и столбца свободных членов  $\bar{b}$  заданы в соответствии со следующей таблицей:

Вариант	Значения матрицы $A$			Значения столбца $\bar{b}$
1	5	0	1	11
	1	3	-1	4
	-3	2	0	6

Вариант	Значения матрицы $A$			Значения столбца $\vec{b}$
2	2	0	-1	-3
	-1	3	1	2
	1	-1	4	3
3	2	0	-1	1
	1	-3	1	2
	1	1	3	1
4	5	1	-1	-5
	-1	3	1	5
	1	-2	4	1
5	3	1	-1	-1
	-2	4	1	5
	1	1	3	3
6	3	1	-1	6
	2	4	1	9
	1	-1	3	4
7	2	-1	0	-2
	2	5	2	-4
	1	-1	3	2
8	3	-1	-1	1
	0	2	-1	3
	-1	1	5	-5
9	4	1	-1	7
	2	3	0	7
	1	-1	5	11
10	2	0	-1	1
	1	-4	2	-5
	1	1	3	6
11	2	-1	0	3
	0	5	2	7
	1	-1	3	4
12	3	1	-1	-2
	1	5	-1	8
	2	0	3	1
13	3	0	-1	-4
	2	-5	1	9
	2	-2	6	8
14	3	0	-1	7
	2	-5	1	-2
	2	2	5	1
15	3	-1	1	0
	3	5	1	12
	-1	2	4	-1
16	-4	2	1	-5
	-1	5	1	-5
	2	0	3	5

Вариант	Значения матрицы $A$	Значения столбца $\bar{b}$
17	5     -1     1 -1     3     1 2     1     4	-3 -1 1
18	4     2     1 3     5     -1 2     1     -4	3 4 6
19	4     1     2 0     3     1 1     2     4	-6 -1 -5
20	3     1     1 0     -2     1 2     -1     4	6 -3 -1
21	2     -1     0 2     5     1 2     1     -4	-5 2 -7
22	5     0     1 1     -3     1 3     -1     5	11 3 11
23	5     -1     3 1     -3     1 0     1     2	0 -6 0
24	5     -1     1 -2     5     1 3     -1     5	-6 13 0
25	5     1     -1 -1     3     1 -1     0     2	8 0 -5

### Контрольные вопросы

1. Какая система уравнений называется совместной (несовместной)?
2. Какая система уравнений называется однородной (неоднородной)? Приведите примеры.
3. Какая система уравнений называется вырожденной (невыврожденной)?
4. В каком случае система уравнений имеет единственное решение?
5. Какой вид имеет система уравнений после окончания выполнения «прямого хода» в методе Гаусса?
6. В чем принципиальное различие прямых и итерационных методов решения СЛАУ?
7. Что называют ведущими элементами в методе Гаусса?

8. Сформулируйте условия, при которых метод простых итераций является сходящимся?

## **Лабораторная работа № 2**

### **Решение систем линейных алгебраических уравнений численными методами**

**Цель работы:** изучение численных методов решения СЛАУ и формирование навыков их реализации на ЭВМ.

#### **Содержание работы**

1. Изучить теоретический материал.
2. Выбрать индивидуальное задание согласно варианту, найти точное решение системы линейных уравнений методом Гаусса с помощью программного модуля, разработанного в лабораторной работе № 1.
3. Выполнить вручную расчет первых двух итераций.
4. Создать исходный модуль программы на языке высокого уровня Паскаль, реализующий метод простых итераций решения СЛАУ с заданной точностью  $\varepsilon$  и отладить его.
5. Получить посредством разработанного модуля решение индивидуального задания и сравнить его с точным решением.
6. Оформить отчет.

#### **Содержание отчета**

1. Тема и цель работы, содержание индивидуального задания.
2. Результаты решения СЛАУ методом Гаусса.
3. Блок-схема алгоритма метода простых итераций.
4. Результаты расчета вручную первых двух итераций.
5. Исходный текст программы и результаты ее тестирования.
6. Результаты работы программы согласно индивидуальному заданию.

#### **Варианты индивидуальных заданий**

Найти решение системы линейных алгебраических уравнений вида  $Ax = \bar{b}$  методом простых итераций с точностью  $\varepsilon = 0,001$ , если значения матрицы  $A$  и столбца свободных членов  $\bar{b}$  заданы в соответствии со следующей таблицей:

Вариант	Значения матрицы $A$			Значения столбца $\bar{b}$
1	7,44	-2,46	2,74	-3,05
	5,41	-1,25	2,01	2,57
	1,15	3,81	-0,92	-1,15
2	1,26	4,20	-1,97	4,21
	0,71	-1,91	3,38	-2,00
	2,20	-4,79	3,16	-5,01
3	3,40	2,82	2,82	3,01
	4,18	1,25	0,95	-4,15
	1,71	-3,95	0,25	0,57
4	1,11	-4,83	2,15	-5,01
	1,75	2,16	-5,01	2,25
	2,43	5,52	-3,39	5,21
5	1,08	-3,50	1,90	4,15
	3,01	-0,15	5,41	-1,27
	0,06	-1,70	5,79	13,18
6	5,39	-1,24	2,03	4,98
	2,03	-1,24	-4,72	2,42
	3,18	2,60	-5,67	3,52
7	3,44	-0,60	1,19	3,50
	0,74	-1,90	3,90	0,85
	5,45	1,72	3,14	5,05
8	0,80	-1,61	-3,76	-1,75
	2,19	1,26	-0,94	1,29
	0,93	-2,38	4,02	-2,50
9	2,51	1,23	1,75	12,34
	1,52	2,15	3,14	19,76
	4,23	4,35	2,72	24,33
10	4,42	12,64	-5,77	-1,56
	0,16	-3,82	6,41	4,42
	-7,14	4,98	-8,73	-2,67
11	1,17	4,19	-1,77	4,25
	1,39	-1,45	-4,60	-1,55
	4,06	-5,42	3,88	-6,01
12	1,21	3,49	-0,99	3,75
	1,50	-0,87	4,21	-1,05
	1,28	-3,66	2,68	-3,91
13	1,10	3,81	-0,93	4,20
	2,03	-1,21	4,75	-1,28
	4,26	-5,06	2,93	-4,85
14	1,13	3,92	-2,17	14,21
	2,41	-1,13	4,48	10,52
	4,04	-6,03	2,63	-13,02
15	5,96	1,40	5,03	-7,41
	2,99	1,25	-0,38	4,48
	1,05	-3,57	1,92	-14,14
16	2,36	2,37	2,13	1,48
	2,52	2,40	2,10	1,92
	2,59	2,41	2,06	2,16

Вариант	Значения матрицы $A$			Значения столбца $b$
17	3,43	3,38	3,09	5,52
	4,17	4,00	3,65	6,93
	4,30	4,10	3,67	7,29
18	2,83	2,50	2,08	33,28
	3,00	2,55	2,07	33,59
	3,72	3,21	2,68	43,43
19	6,21	4,52	5,16	23,17
	5,24	7,49	5,34	47,54
	4,16	5,86	4,15	11,06
20	2,53	2,36	1,93	12,66
	3,95	4,11	3,66	21,97
	2,78	2,43	1,94	13,93
21	2,16	1,96	1,56	13,16
	3,55	3,23	2,78	21,73
	4,85	4,74	3,97	29,75
22	2,69	2,47	2,07	19,37
	2,73	2,39	1,92	19,43
	2,93	2,52	2,02	20,80
23	3,72	3,47	3,06	30,74
	4,47	4,10	3,63	36,80
	4,96	4,53	4,13	40,79
24	3,35	2,94	2,53	70,69
	5,41	4,88	4,41	65,38
	3,88	3,30	2,78	81,07
25	4,25	3,84	3,43	86,07
	3,86	3,34	2,87	77,12
	5,40	4,82	4,30	108,97

### Контрольные вопросы

1. Дайте общую характеристику методов решения систем линейных уравнений.
2. Какие условия являются достаточными для сходимости метода итераций?
3. Что является критерием останова при решении систем линейных уравнений методом итераций?
4. В каком случае система уравнений имеет множество решений?
5. Охарактеризуйте понятие «норма матрицы». Приведите примеры.

### Лабораторная работа № 3

#### Вычисление собственных значений и векторов матриц

**Цель работы:** изучение теоретического материала и освоение методы численного решения задачи вычисления собственных значений и векторов матрицы.

### Содержание работы

1. Изучить теоретический материал.
2. Локализовать собственные значения матрицы, указанной в варианте задания.
3. Определить точные значения собственных чисел и векторов матрицы.
4. Составить блок-схему и программу для вычисления с заданной точностью наибольшего по абсолютной величине собственного числа и соответствующего собственного вектора, используя степенной метод (итерации)
5. Оформить отчет.

### Содержание отчета

1. Сформулировать решаемую задачу и изложить используемый способ её решения.
2. Привести результаты локализации значений собственных чисел матрицы и точные значения собственных чисел и векторов.
3. Привести результаты численных расчетов и сравнить их с точными.

### Варианты индивидуальных заданий

Для данной матрицы  $A$  найти максимальное по абсолютной величине собственное значение (с точностью  $\varepsilon = 0,001$ ) и соответствующий ему собственный вектор:

Вариант 1	Вариант 11	Вариант 21
5    6    -3 -1    0    1 1    2    -1	3    -4    4 1    -1    -8 0    0    -2	6    0    2 5    -1    1 -3    1    -1
Вариант 2	Вариант 12	Вариант 22
2    -1    2 5    -3    3 -1    0    -2	9    -1    8 22    -4    16 -6    1    -5	-1    -3    0 2    5    -1 2    3    -2



Вариант 3	Вариант 13	Вариант 23
0 0 1 0 1 0 1 0 0	1 2 -1 -1 0 1 5 6 -3	5 6 -3 1 2 -1 -1 0 1
Вариант 4	Вариант 14	Вариант 24
0 2 1 -2 0 3 -1 -3 0	-1 0 -2 5 -3 3 2 -1 2	0 2 1 -2 0 3 -1 -3 0
Вариант 5	Вариант 15	Вариант 25
3 1 0 -4 -1 0 4 -8 -2	1 0 1 0 1 0 0 0 1	3 1 0 1 2 2 0 1 1
Вариант 6	Вариант 16	Вариант 26
9 22 -6 -1 -4 1 8 16 -5	-1 -3 0 -2 0 3 0 2 1	0 1 1 1 2 2 3 1 0
Вариант 7	Вариант 17	Вариант 27
5 -1 1 6 0 2 -3 1 -1	4 -8 -2 -4 -1 0 3 1 0	1 2 -4 3 -1 2 1 0 -1
Вариант 8	Вариант 18	Вариант 28
2 5 -1 -1 -3 0 2 3 -2	8 16 -5 -1 -4 1 9 22 -6	1 2 2 1 0 3 0 1 3
Вариант 9	Вариант 19	Вариант 29
0 0 1 0 1 0 1 0 1	4 1 0 1 2 1 0 1 1	2 5 7 6 3 4 5 -2 -3
Вариант 10	Вариант 20	Вариант 30
0 -2 -1 2 0 -3 1 3 0	0 0 1 1 0 1 1 1 0	4 1 0 1 2 1 0 1 1

### Контрольные вопросы

1. Из какого уравнения находятся собственные значения матрицы?
2. Как формулируется проблема собственных значений?
3. Что такое нормировка вектора?
4. Каким свойством обладают действительные и различные собственные значения?
5. Как проводится локализация собственных значений матрицы?

## Лабораторная работа № 4

### Интерполирование функций

**Цель работы:** найти значения функций с помощью интерполяционных многочленов Ньютона.

#### Содержание работы

1. Изучить теоретический материал.
2. Выбрать индивидуальное задание согласно варианту, вычислить вручную значения функции для точки  $a_1$  с применением интерполяционного многочлена Ньютона.
3. Создать исходный модуль программы на языке высокого уровня Паскаль, реализующий вычисление значений функции при заданных значениях аргумента.
4. Оформить отчет.

#### Содержание отчета

1. Тема и цель работы, содержание индивидуального задания.
2. Результаты расчета вручную с применением квадратичной интерполяции (получение значения функции для точки  $a_1$ ).
3. Блок-схема алгоритма.
4. Исходный текст программы.
5. Результаты работы программы согласно индивидуальному заданию.

#### Варианты индивидуальных заданий

Вычислить значения функции  $f(x)$  в точках  $a_1 = \frac{x_0 + x_1}{2}$ ,  $a_2 = \frac{x_1 + x_2}{2}$ ,  $a_3 = \frac{x_2 + x_3}{2}$  с использованием интерполяционного многочлена Ньютона для конечных и разделенных разностей.

Вариант № 1		Вариант № 2		Вариант № 3		Вариант № 4	
$x$	$f(x)$	$x$	$f(x)$	$x$	$f(x)$	$x$	$f(x)$
1,1	0,8912	2,1	0,8632	2,1	0,7481	1,6	1,2359
1,2	0,9320	2,2	0,8085	2,6	0,9841	2,0	1,5645
1,3	0,9636	2,3	0,7457	3,1	1,2037	2,4	2,0063
1,4	0,9854	2,4	0,6755	3,6	1,3454	2,8	2,4112

Вариант № 5		Вариант № 6		Вариант № 7		Вариант № 8	
$x$	$f(x)$	$x$	$f(x)$	$x$	$f(x)$	$x$	$f(x)$
1,15	0,917	2,0	0,5632	1,1	0,741	1,6	1,4359
1,20	0,380	2,2	0,2485	1,6	0,841	2,0	1,3645
1,25	0,666	2,4	0,4657	2,1	1,037	2,4	2,5063
1,30	0,884	2,6	0,5555	2,6	1,354	2,8	2,9112
Вариант № 9		Вариант № 10		Вариант № 11		Вариант № 12	
$x$	$f(x)$	$x$	$f(x)$	$x$	$f(x)$	$x$	$f(x)$
0,1	0,812	2,1	0,632	4,1	0,9481	1,65	1,2459
0,2	0,920	2,2	0,885	4,6	0,9841	1,70	1,3645
0,3	0,636	2,3	0,757	5,1	1,0037	1,75	2,3063
0,4	0,854	2,4	0,755	5,6	1,2454	1,80	2,4662
Вариант № 13		Вариант № 14		Вариант № 15		Вариант № 16	
$x$	$f(x)$	$x$	$f(x)$	$x$	$f(x)$	$x$	$f(x)$
1,1	5,8912	1	4,8632	2,1	0,7581	1,6	1,4359
1,2	4,9320	2	3,8085	2,6	0,9541	2,6	1,4645
1,3	3,9636	3	4,7457	3,1	1,2537	3,6	2,4063
1,4	2,9854	4	5,6755	3,6	1,3554	4,6	2,4112
Вариант № 17		Вариант № 18		Вариант № 19		Вариант № 20	
$x$	$f(x)$	$x$	$f(x)$	$x$	$f(x)$	$x$	$f(x)$
1,15	0,4912	2,1	0,5632	0,1	2,7481	1,6	1,6659
1,25	0,5320	2,2	0,4885	0,3	3,9841	1,7	1,9945
1,35	0,6636	2,3	0,8957	0,5	1,2537	1,8	2,0063
1,45	0,3854	2,4	0,3655	0,7	1,3954	1,9	2,4562
Вариант № 21		Вариант № 22		Вариант № 23		Вариант № 24	
$x$	$f(x)$	$x$	$f(x)$	$x$	$f(x)$	$x$	$f(x)$
1,1	0,4912	2,5	0,8722	2,1	0,6481	1,6	1,2559
1,2	0,6320	3,0	0,1085	2,6	0,5841	2,0	1,5365
Вариант № 21		Вариант № 22		Вариант № 23		Вариант № 24	
$x$	$f(x)$	$x$	$f(x)$	$x$	$f(x)$	$x$	$f(x)$
1,3	0,8636	3,5	0,0457	3,1	1,0037	2,4	2,1263
1,4	1,9854	4,0	0,2755	3,6	1,0454	2,8	2,0212
Вариант № 25		Вариант № 26		Вариант № 27		Вариант № 28	
$x$	$f(x)$	$x$	$f(x)$	$x$	$f(x)$	$x$	$f(x)$
1,17	0,912	2,1	0,6322	2,1	0,4812	1,6	1,359
1,2	0,320	2,25	0,8051	2,65	0,9413	2,0	1,645
1,23	0,966	2,4	0,7471	3,2	1,2376	2,4	2,063
1,26	0,984	2,55	0,6552	3,75	1,3544	2,8	2,412

### Контрольные вопросы

1. Постановка задачи интерполирования функции.
2. Что такое конечная разность  $n$ -го порядка? Приведите пример разности первого и второго порядка функции.
3. Что такое разделенная разность  $n$ -го порядка? Приведите пример разности первого и второго порядка функции.

4. Общий вид интерполяционного многочлена Ньютона для конечных разностей.
5. Общий вид интерполяционного многочлена Ньютона для разделенных разностей.
6. Что такое линейная интерполяция?
7. Перечислите другие методы (не применяемые в работе) интерполяции функции.
8. Оценка погрешности интерполяционной формулы Ньютона.

## Лабораторная работа № 5

### Дифференцирование функций

**Цель работы:** найти значения производных первого и второго порядка численными методами.

#### Содержание работы

1. Изучить теоретический материал.
2. Вычислить первую и вторую производную во внутренних точках  $x_1, x_2$  с помощью центральных разностей; вычислить первую и вторую производную в граничных точках  $x_0, x_3$ , если функция  $y = f(x)$  задана таблично:

$x_0$	$x_1$	$x_2$	$x_3$
$y_0$	$y_1$	$y_2$	$y_3$

#### Содержание отчета

1. Тема и цель работы, содержание индивидуального задания.
2. Таблица значений первой производной.
3. Таблица значений второй производной.

#### Варианты индивидуальных заданий

Вариант №1		Вариант №2		Вариант №3		Вариант №4	
x	$f(x)$	x	$f(x)$	x	$f(x)$	x	$f(x)$
1,1	0,8912	2,1	0,8632	2,1	0,7481	1,6	1,2359
1,2	0,9320	2,2	0,8085	2,6	0,9841	2,0	1,5645
1,3	0,9636	2,3	0,7457	3,1	1,2037	2,4	2,0063
1,4	0,9854	2,4	0,6755	3,6	1,3454	2,8	2,4112

Вариант №5		Вариант №6		Вариант №7		Вариант №8	
x	$f(x)$	x	$f(x)$	x	$f(x)$	x	$f(x)$
1,15	0,917	2,0	0,5632	1,1	0,741	1,6	1,4359
1,20	0,380	2,2	0,2485	1,6	0,841	2,0	1,3645
1,25	0,666	2,4	0,4657	2,1	1,037	2,4	2,5063
1,30	0,884	2,6	0,5555	2,6	1,354	2,8	2,9112

Вариант №9		Вариант №10		Вариант №11		Вариант №12	
x	$f(x)$	x	$f(x)$	x	$f(x)$	x	$f(x)$
0,1	0,812	2,1	0,632	4,1	0,9481	1,65	1,2459
0,2	0,920	2,2	0,885	4,6	0,9841	1,70	1,3645
0,3	0,636	2,3	0,757	5,1	1,0037	1,75	2,3063
0,4	0,854	2,4	0,755	5,6	1,2454	1,80	2,4662

Вариант №13		Вариант №14		Вариант №15		Вариант №16	
x	$f(x)$	x	$f(x)$	x	$f(x)$	x	$f(x)$
1,1	5,8912	1	4,8632	2,1	0,7581	1,6	1,4359
1,2	4,9320	2	3,8085	2,6	0,9541	2,6	1,4645
1,3	3,9636	3	4,7457	3,1	1,2537	3,6	2,4063
1,4	2,9854	4	5,6755	3,6	1,3554	4,6	2,4112

Вариант №17		Вариант №18		Вариант №19		Вариант №20	
x	$f(x)$	x	$f(x)$	x	$f(x)$	x	$f(x)$
1,15	0,4912	2,1	0,5632	0,1	2,7481	1,6	1,6659
1,25	0,5320	2,2	0,4885	0,3	3,9841	1,7	1,9945
1,35	0,6636	2,3	0,8957	0,5	1,2537	1,8	2,0063
1,45	0,3854	2,4	0,3655	0,7	1,3954	1,9	2,4562

Вариант №21		Вариант №22		Вариант №23		Вариант №24	
x	$f(x)$	x	$f(x)$	x	$f(x)$	x	$f(x)$
1,1	0,4912	2,5	0,8722	2,1	0,6481	1,6	1,2559
1,2	0,6320	3,0	0,1085	2,6	0,5841	2,0	1,5365
1,3	0,8636	3,5	0,0457	3,1	1,0037	2,4	2,1263
1,4	1,9854	4,0	0,2755	3,6	1,0454	2,8	2,0212

Вариант №25		Вариант №26		Вариант №27		Вариант №28	
x	$f(x)$	x	$f(x)$	x	$f(x)$	x	$f(x)$
1,17	0,912	2,1	0,6322	2,1	0,4812	1,6	1,359
1,2	0,320	2,2	0,8051	2,6	0,9413	2,0	1,645
1,23	0,966	2,3	0,7471	3,1	1,2376	2,4	2,063
1,26	0,984	2,4	0,6552	3,6	1,3544	2,8	2,412

### Контрольные вопросы

1. Постановка задачи численного дифференцирования.

2. Что такое центральная разность 2-ого порядка? Приведите пример.
3. Что называется погрешностью аппроксимации производной?
4. Использование ряда Тейлора для численного дифференцирования.

### **Лабораторная работа № 6**

#### **Приближенные методы вычисления собственных интегралов**

**Цель работы:** изучить методы приближенного вычисления определенных интегралов

#### **Содержание работы**

1. Изучить теоретический материал.
2. Выбрать индивидуальное задание согласно варианту, вычислить вручную точное значение определенных интегралов.
3. Создать исходный модуль программы на языке высокого уровня Паскаль, реализующий приближенное вычисление определенных интегралов заданными методом с заданной точностью.
4. Оформить отчет.

#### **Содержание отчета**

1. Тема и цель работы, содержание индивидуального задания.
2. Результаты ручного вычисления определенного интеграла.
3. Блок-схема алгоритма.
4. Исходный текст программы.
5. Результаты работы программы согласно индивидуального задания.
6. Оценка погрешности вычислений.

#### **Варианты индивидуальных заданий**

Вычислить значение определенного интеграла  $\int_a^b f(x)dx$  с точностью  $\varepsilon = 0,001$  заданным методом:

Вариант	Интеграл	Метод
1	а) $\int_0^3 3^x dx$ б) $\int_1^3 e^{\sin x} 2 \cos x dx$	Прямоугольников  Трапеций
2	а) $\int_1^3 \frac{dx}{x}$ б) $\int_1^3 e^{3 \sin 3x} \cos 3x dx$	Трапеций  Симпсона
3	а) $\int_0^3 (3x^4 - x^3 + 2x - 5) dx$ б) $\int_1^3 e^{x^{\frac{3}{2}}} \sqrt{x} dx$	Симпсона  Прямоугольников
4	а) $\int_0^3 e^x dx$ б) $\int_1^3 5 \cos x \sin x dx$	Прямоугольников  Симпсона
5	а) $\int_0^3 \frac{dx}{3x-1}$ б) $\int_1^3 3e^{x^2} x dx$	Прямоугольников  Трапеций
6	а) $\int_{\pi/4}^{\pi/3} \operatorname{tg} x dx$ б) $\int_1^2 \frac{x^2 dx}{1+x^6}$	Трапеций  Симпсона

Вариант	Интеграл	Метод
7	а) $\int_5^7 \sqrt[5]{x-1} dx$ б) $\int_1^3 2x \cos x dx$	Симпсона  Прямоугольников
8	а) $\int_0^3 (4^x + 1) dx$ б) $\int_1^3 4x \sin x dx$	Трапеций  Прямоугольников
9	а) $\int_0^3 \frac{dx}{1+x^2}$ б) $\int_1^3 \frac{\sin(\ln x)}{x} dx$	Симпсона  Трапеций
10	а) $\int_2^3 \frac{3x-1}{3x+1} dx$ б) $\int_1^3 x^2 \cos x^3 dx$	Трапеций  Прямоугольников
11	а) $\int_2^7 \frac{5x-1}{5x+1} dx$ б) $\int_1^3 x^2 \operatorname{tg} x^3 dx$	Симпсона  Трапеций
12	а) $\int_{\pi/4}^{\pi/3} (\operatorname{ctg} x + 1) dx$ б) $\int_1^2 \frac{3x^2+1}{1+x^6} dx$	Трапеций  Прямоугольников



Вариант	Интеграл	Метод
13	а) $\int_0^3 \frac{3+x}{1+x^2} dx$ б) $\int_1^3 \frac{\operatorname{tg}(\ln x)}{x} dx$	Симпсона  Прямоугольников
14	а) $\int_0^3 \frac{5}{1+x^2} dx$ б) $\int_8^{10} \frac{\cos(\ln x)}{x} dx$	Симпсона  Трапеций
15	а) $\int_2^6 (3x^2 + 7x - 10 + 2^x) dx$ б) $\int_1^3 \frac{\sin(\ln x)}{x} dx$	Трапеций  Прямоугольников
16	а) $\int_0^1 (7^{x+1} - 1) dx$ б) $\int_1^3 (2 \frac{\sin(\ln x)}{x} - 1) dx$	Симпсона  Прямоугольников
17	а) $\int_0^2 (3 - \sqrt[3]{x} + 2^x) dx$ б) $\int_5^8 \frac{dx}{\sqrt{x-1}}$	Симпсона  Трапеций
18	а) $\int_0^3 (x+3)^{-1} \ln x dx$ б) $\int_1^3 x e^{2x^2} dx$	Трапеций  Прямоугольников

Вариант	Интеграл	Метод
19	а) $\int_1^e \frac{dx}{x(1+\ln^2 x)}$ б) $\int_{\pi/12}^{\pi/4} 3 \sin x dx$	Симпсона  Трапеций
20	а) $\int_0^1 \frac{dx}{1+5x^2}$ б) $\int_0^3 (3x^3 + 2x^2 - x + 12) dx$	Симпсона  Прямоугольников
21	а) $\int_0^e x e^{-x^2} dx$ б) $\int_1^2 (3^{x-3} + 4x) dx$	Трапеций  Прямоугольников
22	а) $\int_{\pi/8}^{\pi/3} \operatorname{ctg}(2x-3) dx$ б) $\int_1^3 \sqrt[3]{x-3} dx$	Симпсона  Трапеций
23	а) $\int_0^1 (\cos 2x + \sin 3x) dx$ б) $\int_3^5 (3^{2x-1} + \sqrt[3]{x-2}) dx$	Симпсона  Трапеций
24	а) $\int_0^{\pi} \frac{x dx}{1+3x^2}$ б) $\int_2^3 \frac{dx}{\sqrt[3]{x-1}}$	Симпсона  Прямоугольников

Вариант	Интеграл	Метод
25	а) $\int_0^1 \frac{dx}{1+7x^2}$ б) $\int_1^3 (\sin 3x - \operatorname{tg}(2x-1)) dx$	Трапеций  Прямоугольников
26	а) $\int_0^1 \frac{dx}{(x+1)^2}$ б) $\int_0^{\pi} (\sin 3x - \sqrt[5]{x+3}) dx$	Симпсона  Прямоугольников
27	а) $\int_0^1 (5^x - x^3) dx$ б) $\int_1^2 \frac{\ln^3 x}{x} dx$	Трапеций  Прямоугольников
28	а) $\int_0^1 \frac{x dx}{1+2x^2}$ б) $\int_1^{\pi} (\cos 2x - \operatorname{tg}(x-1)) dx$	Симпсона  Прямоугольников
29	а) $\int_0^1 (6^x - \sin 2x) dx$ б) $\int_1^3 (3x^3 + 2x^2 + 4) dx$	Трапеций  Прямоугольников
30	а) $\int_{-1}^1 \frac{dx}{x^2}$ б) $\int_0^2 \frac{dx}{\sqrt[3]{(2x-3)^2}}$	Симпсона  Прямоугольников

### Контрольные вопросы

1. В чем заключается геометрический смысл определенного интеграла?
2. Постановка задачи приближенного вычисления определенного интеграла.
3. Формула Ньютона–Лейбница.
4. Формула трапеций.
5. Формула прямоугольников.
6. Формула Симпсона.
7. Как влияет шаг интегрирования на точность вычислений?
8. Каким выражением определяется точность формулы трапеций? Симпсона?

### Лабораторная работа № 7

#### Аппроксимация функций методом наименьших квадратов

**Цель работы:** приобрести навыки при обработке экспериментальных данных; алгоритмизация метода наименьших квадратов; программирование метода наименьших квадратов.

#### Содержание работы

1. Составить алгоритм и реализовать программу для определения на ЭВМ коэффициентов  $a_0, a_1, a_2$ , аппроксимирующей функции  $y = a_0 + a_1\varphi_1(x) + a_2\varphi_2(x)$ , где функции  $\varphi_1(x), \varphi_2(x)$  заданы для каждого варианта.

2. Нанести в одной системе координат различными цветами точки, соответствующие экспериментальным данным  $(x_i, y_i)$ , и точки  $x_i, \hat{y}_i$ ,

где  $\hat{y}_i = a_0 + a_1\varphi_1(x_i) + a_2\varphi_2(x_i)$ .

4. Оформить отчет.

#### Содержание отчета

1. Тема и цель работы, содержание индивидуального задания.
2. Блок-схема алгоритма.
3. Исходный текст программы и результаты ее работы.
4. Графическое изображение экспериментальных данных и полученных с помощью аппроксимирующей функции.

## Варианты индивидуальных заданий

Вариант	$\varphi_1(x)$	$\varphi_2(x)$	$x_i$	$y_i$	Вариант	$\varphi_1(x)$	$\varphi_2(x)$	$x_i$	$y_i$
1	$x$	$\frac{1}{x+1}$	0,2 0,4 0,6 0,8 1	5,1 4,75 4,53 4,4 4,46	2	$\sqrt{x}$	$\ln x$	1 1,2 1,4 1,6 1,8	11,9 12,3 12,5 13,1 13,3
3	$\frac{1}{x}$	$x^2$	0,1 0,2 0,1 0,6 0,8	10,1 5,2 2,53 1,1 1,32	4	$x^2$	$\cos x$	0 0,2 0,4 0,6 0,8	2,1 2,01 2,30 2,71 3,25
5	$x$	$\sqrt{x}$	0,1 0,2 0,3 0,4 0,5	-1,1 1,63 1,82 2,10 3,19	6	$x$	$\frac{1}{x}$	0,2 0,1 0,6 0,8 1,0	6,3 3,2 1,62 1,3 0,9
7	$e^x$	$\ln x$	1 2 3 4 5	-0,7 -4,5 -16,8 -50,1 -110,	8	$\sin x$	$\cos x$	1 1,1 1,2 1,3 1,4	1,3 1,4 1,32 1,2 1,08
9	$\sqrt{x}$	$x$	1 2 3 4 5	4,02 5,31 6,7 8,1 9,8	10)	$\ln x$	$x$	1 2 3 4 5	5,1 6,6 8,1 9,4 11,1
11	$x^2$	$x$	0 0,2 0,3 0,1 0,5	1,59 1,74 2,01 2,1 2,18	12	$x^3$	$x^2$	3 4 5 6 7	27,02 71,01 130,9 243,05 382,9
13	$x^2$	$\sqrt{x}$	1 2 3 4 5	-1,01 7,3 12,7 19,8 30	14	$x$	$\sqrt[3]{x}$	7 8 9 10 11	1,91 3,01 4,08 5,15 6,22
15	$\sqrt{x}$	$\ln x$	1 4 9 16 25	-1,1 0,6 3,1 4,7 6,2	16	$x^3$	$tgx$	0 0,2 0,4 0,6 0,8	2,01 2,1 2,2 2,3 2,41

Вариант	$\varphi_1(x)$	$\varphi_2(x)$	$x_i$	$y_i$	Вариант	$\varphi_1(x)$	$\varphi_2(x)$	$x_i$	$y_i$
17	$\lg x$	$\frac{1}{x}$	1 3 5 7 9	0,02 0,43 0,82 1,08 1,32	18	$x$	$\lg x$	1 2 3 4 5	1 2,6 4 5,2 6,4
19	$\ln^2 x$	$\frac{1}{x}$	1 2 3 4 5	2,1 2,15 2,4 2,51 2,8	20	$x^2$	$\frac{1}{x}$	0,1 0,2 0,4 0,6 0,8	9,1 4,2 1,53 0,4 0,32
21	$\sqrt[3]{x}$	$x$	7 8 9 10 11	8,91 10,01 11,08 12,15 13,22	22	$e^x$	$x^2$	1 1,2 1,4 1,6 1,8	5,6 6,7 7,9 9,4 10,2
23	$\sqrt{x}$	$x$	1 2 3 4 5	3,1 4,42 5,73 7,01 8,26	24	$\cos x$	$\sin x$	1 1,1 1,2 1,3 1,4	0,37 0,45 0,27 0,21 0,15
25	$x^2$	$\sin x$	0,2 0,4 0,6 0,8 1	1,23 1,36 1,55 1,8 2,2	26	$x$	$\sin x$	1 1,1 1,2 1,3 1,4	0,86 2,01 2,12 2,27 2,38
27	$\frac{1}{x}$	$\lg x$	1 3 5 7 9	1,02 1,43 1,82 2,08 2,32	28	$\frac{1}{x}$	$x$	0,1 0,2 0,4 0,6 0,8	10,01 5,2 2,53 1,4 1,3
29	$\sin x$	$x$	0 0,1 0,2 0,3 0,4	1,67 1,77 1,89 2,1 2,12	30	$x^2$	$x^2$	2 3 4 5 6	-6,9 7,02 61,01 120,9 133,1

### Контрольные вопросы

1. Каким свойством обладают функции  $\varphi_1(x)$  и  $\varphi_2(x)$ ?
2. На основании каких соотношений составляется система для нахождения коэффициентов  $a_0, a_1, a_2$ ?
3. Что такое невязка?

## Лабораторная работа № 8

### Приближенные методы решения нелинейных уравнений и систем нелинейных уравнений

**цель работы:** изучение методов численного решения нелинейных уравнений и систем нелинейных уравнений

#### Содержание работы

1. Изучить теоретический материал.
2. Выбрать индивидуальное задание согласно варианту, выполнить отделение корней аналитическим или графическим методом.
3. Создать исходный модуль программы на языке высокого уровня Паскаль, реализующий вычисление корней нелинейного уравнения и решения систем нелинейных уравнений заданным методом с точностью  $\varepsilon = 0.001$ .
4. Оформить отчет.

#### Содержание отчета

1. Тема и цель работы, содержание индивидуального задания.
2. Результаты отделения корней уравнения.
3. Блок-схема алгоритма.
4. Исходный текст программы.
5. Результаты работы программы согласно индивидуальному заданию.

#### Варианты индивидуальных заданий

Вариант	Уравнение и система уравнений	Метод решения уравнения и системы уравнений
1	а) $2x + 1 - \sin x = 0$	Метод итерации
	б) $\begin{cases} y(x-1) - 1 = 0 \\ x^2 - y^2 = 0 \end{cases}$	Метод Ньютона
2	а) $1 - x + \cos x = 0$	Метод касательных
	б) $\begin{cases} y - x^2 + 0,2 = 0 \\ x - y^2 = 0 \end{cases}$	Метод Ньютона

Вариант	Уравнение и система уравнений	Метод решения уравнения и системы уравнений
3	а) $(x+1)\cos x - 1 = 0$	Метод половинного деления
	б) $\begin{cases} y - \ln x = 0 \\ x^2 + y^2 - 3 = 0 \end{cases}$	Метод Ньютона
4	а) $(x-1)\sin x + 1 = 0$	Метод половинного деления
	б) $\begin{cases} y^2 + x^2 - 3 = 0 \\ x - y^2 = 0 \end{cases}$	Метод Ньютона
5	а) $x^2 - \cos x = 0$	Метод касательных
	б) $\begin{cases} x^2 + 4y^2 - 4 = 0 \\ x^2 - 4x + 36y^2 - 24y + 4 = 0 \end{cases}$	Метод Ньютона
6	а) $\ln x - 1 + x = 0$	Метод касательных
	б) $\begin{cases} y - 0,8\cos(x-1) = 0 \\ e^{-y} - x = 0 \end{cases}$	Метод итераций
7	а) $\lg x + x^2 = 0$	Метод половинного деления
	б) $\begin{cases} 2y + x - 1 = 0 \\ 0,2y^2 + x - 1 = 0 \end{cases}$	Метод Ньютона
8	а) $0,5e^x + x = 0$	Метод касательных
	б) $\begin{cases} y - 0,5\sin x - 1 = 0 \\ y + 2x - 1 = 0 \end{cases}$	Метод итерации
9	а) $\sqrt{x+1} - \frac{1}{x} = 0$	Метод половинного деления
	б) $\begin{cases} \sqrt{x} + y = 0 \\ 4x^2 + y^2 - 4 = 0 \end{cases}$	Метод Ньютона
10	а) $x^3 - 3x + 2 = 0$	Метод касательных
	б) $\begin{cases} y - 0,5\lg x - 1 = 0 \\ y^2 + 10x - 10 = 0 \end{cases}$	Метод итерации
11	а) $5 - x^2 - e^x = 0$	Метод хорд
	б) $\begin{cases} y + \sqrt{x} = 0 \\ y - \ln x = 0 \end{cases}$	Метод итерации



Вариант	Уравнение и система уравнений	Метод решения уравнения и системы уравнений
12	а) $7 - x - e^x = 0$ б) $\begin{cases} y - \ln x - 1 = 0 \\ 0,8 \cos(y - 1,7) - x + 1,2 = 0 \end{cases}$	Метод итераций Метод Ньютона
13	а) $\cos 2x - x^2 = 0$ б) $\begin{cases} y - \operatorname{tg} x = 0 \\ 1,5y - \sqrt{2,25 - x^2} = 0 \end{cases}$	Метод касательных Метод Ньютона
14	а) $x^3 + 2x = 3 = 0$ б) $\begin{cases} 0,5y - x + 1 = 0 \\ y + x^2 - 2x + 2 = 0 \end{cases}$	Метод касательных Метод итерации
15	а) $\sin x - \frac{1}{x+3} = 0$ б) $\begin{cases} 2y + x - 7 = 0 \\ x^2 + 4y^2 - 25 = 0 \end{cases}$	Метод половинного деления Метод Ньютона
16	а) $1 - x^2 + \sin x = 0$ б) $\begin{cases} x - \sqrt{2 - y} - 2 = 0 \\ 5y + x = 0 \end{cases}$	Метод половинного деления Метод итерации
17	а) $2 - x^2 + \cos x = 0$ б) $\begin{cases} 2x - y - 10 = 0 \\ 5x^2 - 20y^2 - 100 = 0 \end{cases}$	Метод касательных Метод Ньютона
18	а) $1 - x + \sin x = 0$ б) $\begin{cases} \cos y + 4x = 0 \\ 2y + x^2 - \pi = 0 \end{cases}$	Метод касательных Метод итераций
19	а) $2x + 1 + \cos x = 0$ б) $\begin{cases} 5x^2 - 20y^2 + 100 = 0 \\ y - \sqrt{3x} = 0 \end{cases}$	Метод касательных Метод Ньютона

Вариант	Уравнение и система уравнений	Метод решения уравнения и системы уравнений
20	а) $x \cos(x+1) - 1 = 0$ б) $\begin{cases} 2y - 5x - 18 = 0 \\ -4x^2 + y^2 + 36 = 0 \end{cases}$	Метод половинного деления Метод итерации
21	а) $\lg x - 1 + x = 0$ б) $\begin{cases} x - 2y + 1 = 0 \\ 9x^2 - 16y^2 - 144 = 0 \end{cases}$	Метод половинного деления Метод Ньютона
22	а) $0,3 \ln x + x^2 = 0$ б) $\begin{cases} e^y - x^2 = 0 \\ \cos(y - \ln x) - 2x + 7 = 0 \end{cases}$	Метод касательных Метод итерации
23	а) $e^x - \ln x = 0$ б) $\begin{cases} x^2 + y^2 - 2x - 0,21 = 0 \\ y - \ln(x-1) = 0 \end{cases}$	Метод касательных Метод Ньютона
24	а) $\sqrt{x+1} + x - 2 = 0$ б) $\begin{cases} y + yx^2 - 1 = 0 \\ 4x + y^2 - 1 = 0 \end{cases}$	Метод половинного деления Метод итерации
25	а) $x^3 + 3x^2 - 3 = 0$ б) $\begin{cases} 2x - y - 3 = 0 \\ 9x^2 + 16y^2 - 144 = 0 \end{cases}$	Метод половинного деления Метод Ньютона
26	а) $2 \sin x + x - 1 = 0$ б) $\begin{cases} x(1 + y^2) - 1 = 0 \\ x^2 + 6y - 1 = 0 \end{cases}$	Метод хорд Метод итерации
27	а) $x \sin(x-1) - 2 = 0$ б) $\begin{cases} y^3 - x + 1 = 0 \\ x^2 + y - 1 = 0 \end{cases}$	Метод половинного деления Метод итерации

Вариант	Уравнение и система уравнений	Метод решения уравнения и системы уравнений
28	а) $\sin 2x + x^2 - 1 = 0$	Метод касательных
	б) $\begin{cases} y^2 - x^3 = 0 \\ 4y - x + 2 - 4\sqrt{8} = 0 \end{cases}$	Метод итерации
29	а) $x^3 - 2x^2 + 3 = 0$	Метод касательных
	б) $\begin{cases} xy - 2 = 0 \\ 2y - \ln x - 4 = 0 \end{cases}$	Метод Ньютона
Вариант	Уравнение и система уравнений	Метод решения уравнения и системы уравнений
30	а) $\sin x - \frac{1}{x+3} = 0$	Метод половинного деления
	б) $\begin{cases} y - e^{x-1} = 0 \\ 3e^x - e^y - 1 = 0 \end{cases}$	Метод итерации

### Контрольные вопросы

1. Какие условия должны выполняться для сходимости метода Ньютона в случае решения нелинейного уравнения?
2. В чем состоит суть метода простых итераций?
3. Что такое условия Фурье?
4. Какие способы отделения корней вы знаете?

### Лабораторная работа № 9

#### Численные методы решения обыкновенных дифференциальных уравнений

**Цель работы:** изучение методов численного интегрирования обыкновенных дифференциальных уравнений

#### Содержание работы

1. Изучить теоретический материал.
2. Выбрать индивидуальное задание согласно варианту, выполнить вручную расчет первых двух значений дискретной функции, являющейся решением данного дифференциального уравнения.

3. Создать исходный модуль программы на языке высокого уровня Паскаль, реализующий приближенное вычисление дифференциального уравнения заданным методом с заданной точностью.

4. Оформить отчет.

### Содержание отчета

1. Тема и цель работы, содержание индивидуального задания.
2. Результаты ручного расчета первых двух значений приближенного решения.
3. Блок-схема алгоритма.
4. Исходный текст программы.
5. Результаты работы программы согласно индивидуальному заданию.

### Варианты индивидуальных заданий

Решить дифференциальное уравнение с данными начальными условиями на интервале  $[a, b]$  с шагом  $h=0,1$  двумя способами: методом Эйлера и методом Рунге–Кутты четвертого порядка. Произвести оценку полученных решений.

Вариант	Дифференциальное уравнение	Начальное условие	Интервал интегрирования
1	$y' = \frac{1}{2}xy$	$y(0) = 1$	$[0,1]$
2	$y' = x^2 + y^2$	$y(0) = 1$	$[0,1]$
3	$y' = 1 + xy^2$	$y(0) = 1$	$[0,1]$
4	$y' = \frac{y}{x+1} - y^2$	$y(0) = 1$	$[0,1]$
5	$y' = -y^2 + \frac{0,1}{x^2}$	$y(1) = 1$	$[1,2]$
6	$y' = \frac{1}{x^2} - \frac{y}{x} - 2y^2$	$y(1) = 2$	$[1,2]$
7	$y' = 1 + y \sin x - 2y$	$y(0) = 0$	$[0,1]$
8	$y' = y - \frac{2x}{y}$	$y(0) = 1$	$[0,1]$
9	$y' = x^2 - y^2$	$y(0) = 0$	$[0,1]$
10	$y' + y = 5xe^x$	$y(0) = 1$	$[0,1]$

Вариант	Дифференциальное уравнение	Начальное условие	Интервал интегрирования
11	$y' + y = 1 + e^x$	$y(0) = 2.5$	[0,1]
12	$y' = 2x + \sin \frac{y}{x}$	$y(0) = 1$	[0,1]
13	$y' = \sqrt{x} + \sqrt{y}$	$y(0) = 0$	[0,1]
14	$y' = x + \sqrt{3 + y^2}$	$y(0) = 0$	[0,1]
15	$y' + y = 0$	$y(0) = 0$	[0,1]
16	$y' = 2x - 3y$	$y(0) = 0$	[0,1]
17	$y' + y = 5 + x$	$y(0) = 3$	[0,1]
18	$y' = 2x + \sin y$	$y(0) = 1$	[0,1]
19	$y' = \frac{e^2}{x + 2y} + \frac{x}{4}$	$y(1.8) = 4.5$	[1.8,4.6]
20	$y' = -2y$	$y(0) = 1$	[0,1]
21	$y' + y = xe^x$	$y(0) = 1$	[0,1]
22	$y' = 2 + e^x - y$	$y(0) = 1$	[0,1]
23	$y' = 2x - y$	$y(0) = 0$	[0,1]
24	$y' + y = e^{-x} \cos x$	$y(0) = 1$	[0,1]
25	$y' - 2y + y = xe^x$	$y(0) = 1$	[0,1]

### Контрольные вопросы

1. Для чего используются дифференциальные уравнения?
2. Приведите геометрическую интерпретацию метода Эйлера.
3. Какие методы называются одношаговыми? многошаговыми?
4. Почему метод Рунге–Кутты четвертого порядка точнее метода Эйлера?

## Заключение

Настоящий лабораторный практикум содержит лишь малую часть материала из раздела вычислительной математики, без знания которой невозможно представить умелое применение вычислительной техники и хорошего специалиста по вопросам различного планирования, если он не владеет методами приближенного анализа.

## Библиографический список

1. *Турчак, Л. И.* Основы численных методов: учебное пособие/ Л. И. Турчак. – М.:Наука, 1987.
2. *Боглаев, Ю. П.* Вычислительная математика и программирование: учебное пособие для студентов втузов/ Ю. П. Боглаев. – М.: Высшая шк., 1990.
3. *Бахвалов, Н. С.* Численные методы: учебное пособие/ Н. С. Бахвалов, Н. П. Жидков, Г. М. Кобельков – М.: Наука, 1987.
4. *Калиткин, Н. Н.* Численные методы/ Н. Н. Калиткин. – М.: Наука, 1978.
5. *Тихонов, А. Н.* Методы решения некорректных задач/ А. Н. Тихонов, В. Я. Арсенин. – М.: Наука, 1986.
6. *Форсайт, Дж* Машинные методы математических вычислений/ Дж. Форсайт, М. Малькольм, К. Моулер. – М.: Мир, 1980.
7. *Ортега, Дж* Итерационные методы решения нелинейных систем уравнений со многими неизвестными/ Дж. Ортега, В. Рейнболдт. – М.: Мир, 1975.
8. *Островский, А. М.* Решение уравнений и систем уравнений/ А. М. Островский. – М.:Изд.иностр.лит., 1963.
9. *Хемминг, Р. В.* Численные методы/ Р. В. Хемминг. – М.: Наука, 1968.

Учебное издание

**Ушакова** Наталья Николаевна  
**Подгорный** Николай Николаевич  
**Винтаев** Владимир Николаевич

**ВЫЧИСЛИТЕЛЬНАЯ МАТЕМАТИКА**

Лабораторный практикум

Подписано в печать 27.12.10. Формат 60x84/16. Усл. печ. л. 8,7. Уч.-изд.л. 9,3.

Тираж 500 экз. Заказ Цена

Отпечатано в Белгородском государственном технологическом университете  
им. В. Г. Шухова

308012, г. Белгород, ул. Костюкова, 46

