

СОДЕРЖАНИЕ

| | | |
|-------|--|----|
| 1 | Метод Гаусса решения систем линейных уравнений | 4 |
| 1.1 | Прямой ход метода Гаусса | 5 |
| 1.2 | Обратный ход метода Гаусса. | 8 |
| 1.3 | Численное решение системы линейных алгебраических уравнений методом Гаусса | 8 |
| 1.4 | Метод Гаусса с выбором главного элемента | 11 |
| 1.5 | Численное решение системы линейных алгебраических уравнений методом Гаусса с выбором главного элемента по всей матрице | 12 |
| 2 | Интерполирование функций | 15 |
| 2.1 | Линейная интерполяция функции | 15 |
| 2.2 | Интерполяция алгебраическими полиномами | 16 |
| 2.3 | Интерполяционный полином в форме Лагранжа | 17 |
| 2.4 | Интерполяция функции заданной таблично обобщенным полиномом . . | 19 |
| 2.5 | Интерполяция функции заданной таблично алгебраическими полиномами в форме Лагранжа | 21 |
| 3 | Численное дифференцирование | 25 |
| 3.1 | Численного дифференцирование функции заданной таблично | 27 |
| 4 | Численное интегрирование | 37 |
| 4.1 | Формула прямоугольников | 39 |
| 4.2 | Формула трапеций | 40 |
| 4.3 | Формула Симпсона | 41 |
| 4.4 | Численное интегрирования функции заданной таблично | 42 |
| 5 | Решение нелинейных уравнений | 50 |
| 5.1 | Итервальный метод | 51 |
| 5.2 | Метод бисекции | 52 |
| 5.3 | Метод выделения корней. | 55 |
| 5.4 | Численное решение нелинейного уравнения методом бисекции | 56 |
| 6 | Методы локальной оптимизации | 63 |
| 6.1 | Минимум функции одного переменного | 63 |
| 6.2 | Минимизация функций многих переменных | 67 |
| 6.2.1 | Спуск по координатам | 67 |
| 6.2.2 | Метод градиентного спуска | 71 |

| | |
|--|-----|
| 6.2.3 Метод тяжелого шара | 73 |
| 6.3 Численная оптимизация местоположения склада торговой сети | 74 |
| 6.3.1 Метод по координатного спуска | 74 |
| 6.3.2 Метод градиентного спуска | 87 |
| 7 Аппроксимация функция | 95 |
| 7.1 Точечное квадратичное аппроксимирование функций. | 95 |
| 7.2 Аппроксимирования функций полиномом второй степени $p_2(x)$ | 97 |
| 8 Задача Коши для систем обыкновенных дифференциальных уравнений . | 103 |
| 8.1 Метод Эйлера решения задачи Коши. | 104 |
| 8.2 Оценка погрешности решения задачи Коши. | 106 |
| 8.3 Численное решение задачи Коши методом Эйлера | 109 |
| 9 Краевые задачи для обыкновенных дифференциальных уравнений | 116 |
| 9.1 Линейные краевые условия | 118 |
| 9.2 Построение разностных схем методом баланса | 119 |
| 9.2.1 Физическая интерпретация краевых задач | 120 |
| 9.2.2 Расчётная сетка. | 124 |
| 9.2.3 Разностная схема для дифференциального уравнения. | 128 |
| 9.2.4 Разностная аппроксимация краевых условий | 133 |
| 9.3 Метод решения систем линейных уравнений с матрицами специаль- ного вида | 138 |
| 9.3.1 Метод прогонки | 138 |

1 Метод Гаусса решения систем линейных уравнений

К решению систем линейных алгебраических уравнений сводится подавляющее большинство задач вычислительной математики и многие численные методы основаны на решении систем линейных уравнений вида:

$$\left\{ \begin{array}{l} a_{11} \cdot x_1 + a_{12} \cdot x_2 + \dots + a_{1m} \cdot x_m = b_1 \\ a_{21} \cdot x_1 + a_{22} \cdot x_2 + \dots + a_{2m} \cdot x_m = b_2 \\ a_{31} \cdot x_1 + a_{32} \cdot x_2 + \dots + a_{3m} \cdot x_m = b_3 , \\ \vdots \\ a_{m1} \cdot x_1 + a_{m2} \cdot x_2 + \dots + a_{mm} \cdot x_m = b_m \end{array} \right. \quad (1)$$

где x_1, x_2, \dots, x_m – неизвестные, которые необходимо определить; $\{a_{ij}\}$ и b_1, b_2, \dots, b_m – известные числовые коэффициенты левой и правой частей системы уравнений, соответственно.

Решение системы линейных алгебраических уравнений (1) представляет совокупность m действительных или мнимых чисел $\{s_1, s_2, \dots, s_m\}$, таких что их соответствующая подстановка вместо $\{x_1, x_2, \dots, x_m\}$ в систему обращает все её уравнения в тождества:

$$\left\{ \begin{array}{l} a_{11} \cdot s_1 + a_{12} \cdot s_2 + \dots + a_{1m} \cdot s_m \equiv b_1 \\ a_{21} \cdot s_1 + a_{22} \cdot s_2 + \dots + a_{2m} \cdot s_m \equiv b_2 \\ a_{31} \cdot s_1 + a_{32} \cdot s_2 + \dots + a_{3m} \cdot s_m \equiv b_3 \\ \vdots \\ a_{m1} \cdot s_1 + a_{m2} \cdot s_2 + \dots + a_{mm} \cdot s_m \equiv b_m \end{array} \right.$$

Система линейных алгебраических уравнений (1) может быть представлена в более компактной матричной форме:

$$\mathbf{A} \cdot \mathbf{x} = \mathbf{b}, \quad (2)$$

где \mathbf{A} – квадратная матрица $m \times m$, \mathbf{x} и \mathbf{b} – искомый вектор неизвестных и

заданный вектор размерности $1 \times m$, правых частей системы уравнений:

$$\mathbf{A} = \begin{pmatrix} a_{11} & a_{12} & \cdots & a_{1m} \\ a_{21} & a_{22} & \cdots & a_{2m} \\ \vdots & \vdots & \ddots & \vdots \\ a_{m1} & a_{m2} & \cdots & a_{mm} \end{pmatrix}, \quad \mathbf{x} = \begin{pmatrix} x_1 \\ x_2 \\ \vdots \\ x_m \end{pmatrix}, \quad \mathbf{b} = \begin{pmatrix} b_1 \\ b_2 \\ \vdots \\ b_m \end{pmatrix}.$$

Теорема Кронекера–Капелли устанавливает необходимое и достаточное условие совместности системы линейных алгебраических уравнений посредством свойств матричных представлений: система совместна тогда и только тогда, когда ранг её матрицы совпадает с рангом расширенной матрицы, полученной путем добавления столбца правых частей \mathbf{b} матрице системы \mathbf{A} :

$$(\mathbf{A} | \mathbf{b}) = \left(\begin{array}{cccc|c} a_{11} & a_{12} & \cdots & a_{1m} & b_1 \\ a_{21} & a_{22} & \cdots & a_{2m} & b_2 \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ a_{m1} & a_{m2} & \cdots & a_{mm} & b_m \end{array} \right) \quad (3)$$

Преимущество расширенной матрицы заключается в возможности выполнения тех же операций с вектором правых частей системы уравнений, что и со строками матрицы.

Предполагается, что определитель матрицы \mathbf{A} отличен от нуля $\det \mathbf{A} \neq 0$, так что решение $\mathbf{x} = (x_1, x_2, \dots, x_m)^T$ существует и единственно. Систему линейных уравнений можно решить по крайней мере двумя способами: либо воспользовавшись *формулами Крамера*, либо методом последовательного исключения неизвестных (*методом Гаусса*). При больших порядка матрицы m способ Крамера, основанный на вычислении определителей, требует порядка $m!$ арифметических действий, в то время как метод Гаусса – $O(m^3)$ действий.

Для большинства вычислительных задач характерным является большой порядок матрицы \mathbf{A} ($m \approx 10^2 \dots 10^5$), поэтому метод Гаусса в различных вариантах широко используется при решении задач линейной алгебры на ЭВМ.

1.1 Прямой ход метода Гаусса

Метод Гаусса состоит в последовательном исключении неизвестных x_i из системы линейных уравнений (3). Например, предположим, что $a_{11} \neq 0$, тогда

разделим первое уравнение системы на a_{11} , и в результате получим:

$$\left(\begin{array}{cccc|c} 1 & c_{12} & \cdots & c_{1m} & y_1 \\ a_{21} & a_{22} & \cdots & a_{2m} & b_2 \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ a_{m1} & a_{m2} & \cdots & a_{mm} & b_m \end{array} \right),$$

где c_{1j} и y_1 – нормированные коэффициенты 1-ой строки и правой части 1-го уравнения, соответственно:

$$c_{1j} = \frac{a_{1j}}{a_{11}} \quad (j = 2, 3, \dots, m), \quad y_1 = \frac{b_1}{a_{11}}.$$

Последовательно умножим первое уравнение системы на a_{i1} и вычтем полученное уравнение из каждого i -го уравнения системы $i = 2, 3, \dots, m$. В результате получим следующую матрицу:

$$\left(\begin{array}{cccc|c} 1 & c_{12} & \cdots & c_{1m} & y_1 \\ 0 & a_{22} - a_{21} \cdot c_{12} & \cdots & a_{2m} - a_{21} \cdot c_{1m} & b_2 - a_{21} \cdot y_1 \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & a_{m2} - a_{m1} \cdot c_{12} & \cdots & a_{mm} - a_{m1} \cdot c_{1m} & b_m - a_{m1} \cdot y_1 \end{array} \right).$$

Запишем полученную матрицу в более компактном виде:

$$\left(\begin{array}{cccc|c} 1 & c_{12} & \cdots & c_{1m} & y_1 \\ 0 & a_{22}^{(1)} & \cdots & a_{2m}^{(1)} & b_2^{(1)} \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & a_{m2}^{(1)} & \cdots & a_{mm}^{(1)} & b_m^{(1)} \end{array} \right),$$

где $a_{ij}^{(1)}$ и $b_i^{(1)}$ – модифицированные коэффициенты при неизвестных и правой части, соответственно.

$$a_{ij}^{(1)} = (a_{ij} - a_{i1} \cdot c_{1j}), \quad b_i^{(1)} = (b_i - a_{i1} \cdot y_1), \quad (i, j = 2, 3, \dots, m)$$

Если $a_{22}^{(1)} \neq 0$, то в модифицированной системе аналогично можно исключить неизвестное x_2 . Для этого можно разделить второе уравнение системы на

коэффициент при второй неизвестной $a_{22}^{(1)}$, и в результате получим:

$$\left(\begin{array}{cccc|c} 1 & c_{12} & \cdots & c_{1m} & y_1 \\ 0 & 1 & \cdots & c_{2m} & y_2 \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & a_{m2}^{(1)} & \cdots & a_{mm}^{(1)} & b_m^{(1)} \end{array} \right),$$

где c_{2j} и y_2 – нормированные коэффициенты 2-ой строки и правой части 2-го уравнения, соответственно.

$$c_{2j} = \frac{a_{2j}^{(1)}}{a_{22}^{(1)}} \quad (j = 3, 4, \dots, m), \quad y_2 = \frac{b_1^{(1)}}{a_{22}^{(1)}}.$$

Последовательно умножим второе уравнение системы на $a_{i2}^{(1)}$ и вычтем полученное уравнение из каждого i -го уравнения системы $i = 3, 4, \dots, m$. В результате расширенная матрица примет вид:

$$\left(\begin{array}{cccc|c} 1 & c_{12} & \cdots & c_{1m} & y_1 \\ 0 & 1 & \cdots & c_{2m} & y_2 \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & \cdots & a_{mm}^{(1)} - a_{m2}^{(1)} \cdot c_{2m} & b_m^{(1)} - a_{m2}^{(1)} \cdot y_2 \end{array} \right)$$

или в более компактном виде:

$$\left(\begin{array}{cccc|c} 1 & c_{12} & \cdots & c_{1m} & y_1 \\ 0 & 1 & \cdots & c_{2m} & y_2 \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & \cdots & a_{mm}^{(2)} & b_m^{(2)} \end{array} \right)$$

где $a_{ij}^{(2)}$ и $b_i^{(2)}$ – повторно модифицированные коэффициенты при неизвестных и правой части, соответственно:

$$a_{ij}^{(2)} = (a_{ij}^{(1)} - a_{i2}^{(1)} \cdot c_{2j}), \quad b_i^{(2)} = (b_i^{(1)} - a_{i2}^{(1)} \cdot y_2), \quad (i, j = 3, 4, \dots, m)$$

Исключая таким же образом неизвестные x_3, x_4, \dots, x_m , исходная система

линейных уравнений приводится к эквивалентному виду:

$$\left(\begin{array}{cccc|c} 1 & c_{12} & \cdots & c_{1m} & y_1 \\ 0 & 1 & \cdots & c_{2m} & y_2 \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & \cdots & 1 & y_m \end{array} \right) \quad (4)$$

1.2 Обратный ход метода Гаусса

Обратный ход заключается в нахождении неизвестных x_1, x_2, \dots, x_m полученной эквивалентной системы в прямом ходе метода Гаусса. Поскольку расширенная матрица системы (4) имеет треугольный вид, то можно последовательно найти все неизвестные x_m, x_{m-1}, \dots, x_1 :

$$\left\{ \begin{array}{l} x_m = y_m \\ x_{m-1} = y_{m-1} - c_{m-1,m} \cdot x_m \\ x_{m-2} = y_{m-2} - c_{m-2,m-1} \cdot x_{m-1} - c_{m-2,m} \cdot x_m \\ \dots\dots\dots = \dots\dots\dots \\ x_1 = y_1 - \sum_{j=2}^m c_{1j} \cdot x_j \end{array} \right.$$

Общие формулы обратного хода имеют вид:

$$x_m = y_m, \quad x_i = y_i - \sum_{j=i+1}^m c_{ij} \cdot x_j, \quad i = (m-1, m-2, \dots, 1)$$

1.3 Численное решение системы линейных алгебраических уравнений методом Гаусса

Представим систему линейных алгебраических уравнений в матричном виде и запишем расширенную матрицу этой системы, полученную путем до-

бавления к матрице системы столбца правой части уравнений:

$$\begin{cases} 2x_1 + 3x_2 + x_3 = 10 \\ 4x_1 + 5x_2 + 6x_3 = 31 \\ 3x_1 + x_2 + 5x_3 = 22 \end{cases} \iff \left(\begin{array}{ccc|c} 2 & 3 & 1 & 10 \\ 4 & 5 & 6 & 31 \\ 3 & 1 & 5 & 22 \end{array} \right)$$

Прямой ход метода Гаусса.

- 1) Разделим каждую строку матрицы на значение её элемента в первом столбце, т.е. первую строку делим на 2, вторую на 4, третью на 3:

$$\left(\begin{array}{ccc|c} 2 & 3 & 1 & 10 \\ 4 & 5 & 6 & 31 \\ 3 & 1 & 5 & 22 \end{array} \right) \rightarrow \left(\begin{array}{ccc|c} 1 & \frac{3}{2} & \frac{1}{2} & \frac{10}{2} \\ 1 & \frac{5}{4} & \frac{6}{4} & \frac{31}{4} \\ 1 & \frac{1}{3} & \frac{5}{3} & \frac{22}{3} \end{array} \right)$$

Вычитаем из второй и третьей строк матрицы её первую строку:

$$\left(\begin{array}{ccc|c} 1 & \frac{3}{2} & \frac{1}{2} & \frac{10}{2} \\ 1 & \frac{5}{4} & \frac{6}{4} & \frac{31}{4} \\ 1 & \frac{1}{3} & \frac{5}{3} & \frac{22}{3} \end{array} \right) \rightarrow \left(\begin{array}{ccc|c} 1 & \frac{3}{2} & \frac{1}{2} & \frac{10}{2} \\ 0 & -\frac{1}{4} & 1 & \frac{11}{4} \\ 0 & -\frac{7}{6} & \frac{7}{6} & \frac{7}{3} \end{array} \right)$$

- 2) Разделим вторую строку матрицы на $-\frac{1}{4}$, третью строку на $-\frac{7}{6}$:

$$\left(\begin{array}{ccc|c} 1 & \frac{3}{2} & \frac{1}{2} & \frac{10}{2} \\ 0 & -\frac{1}{4} & 1 & \frac{11}{4} \\ 0 & -\frac{7}{6} & \frac{7}{6} & \frac{7}{3} \end{array} \right) \rightarrow \left(\begin{array}{ccc|c} 1 & \frac{3}{2} & \frac{1}{2} & \frac{10}{2} \\ 0 & 1 & -4 & -11 \\ 0 & 1 & -1 & -2 \end{array} \right)$$

Вычитаем из третьей строки матрицы её вторую строку:

$$\left(\begin{array}{ccc|c} 1 & \frac{3}{2} & \frac{1}{2} & \frac{10}{2} \\ 0 & 1 & -4 & -11 \\ 0 & 1 & -1 & -2 \end{array} \right) \rightarrow \left(\begin{array}{ccc|c} 1 & \frac{3}{2} & \frac{1}{2} & \frac{10}{2} \\ 0 & 1 & -4 & -11 \\ 0 & 0 & 3 & 9 \end{array} \right)$$

3) Разделим третью строку матрицы на 3:

$$\left(\begin{array}{ccc|c} 1 & \frac{3}{2} & \frac{1}{2} & \frac{10}{2} \\ 0 & 1 & -4 & -11 \\ 0 & 0 & 3 & 9 \end{array} \right) \rightarrow \left(\begin{array}{ccc|c} 1 & \frac{3}{2} & \frac{1}{2} & \frac{10}{2} \\ 0 & 1 & -4 & -11 \\ 0 & 0 & 1 & 3 \end{array} \right)$$

Обратный ход метода Гаусса.

Последовательно определяем неизвестные в обратном порядке их следования $x_3 \rightarrow x_2 \rightarrow x_1$:

1) Из третьего уравнения системы (третья строка матрицы) определяем неизвестное x_3 :

$$x_3 = 3$$

2) Из второго уравнения системы (вторая строка матрицы) определяем неизвестное x_2 :

$$\begin{aligned} x_2 - 4x_3 &= -11 \rightarrow x_2 = 4x_3 - 11 \\ x_2 &= 4 \cdot 3 - 11 = 1 \end{aligned}$$

3) Из первого уравнения системы (первая строка матрицы) определяем неизвестное x_1 :

$$\begin{aligned} x_1 + \frac{3}{2}x_2 + \frac{1}{2}x_3 &= \frac{10}{2} \rightarrow x_1 = -\frac{3}{2}x_2 - \frac{1}{2}x_3 + 5 \\ x_1 &= -\frac{3}{2} \cdot 1 - \frac{1}{2} \cdot 3 + 5 = 2 \end{aligned}$$

Таким образом, найдено решение системы:

$$\mathbf{s} = \begin{pmatrix} 2 \\ 1 \\ 3 \end{pmatrix} \Leftrightarrow \mathbf{s} = (2, 1, 3)^T$$

Проведём *проверку решения* системы уравнений методом прямой подстановки найденного вектора неизвестных $\mathbf{s} = (2, 1, 3)^T$ в исходную систему урав-

нений:

$$\mathbf{A} \cdot \mathbf{s} = \mathbf{b} \quad \Longleftrightarrow \quad \left(\begin{array}{ccc|c} 2 & 3 & 1 & 10 \\ 4 & 5 & 6 & 31 \\ 3 & 1 & 5 & 22 \end{array} \right) \cdot \begin{pmatrix} 2 \\ 1 \\ 3 \end{pmatrix}$$

После умножения расширенной матрицы системы уравнений на вектор найденного решения получим тождество:

$$\left(\begin{array}{ccc|c} 2 \cdot 2 + 3 \cdot 1 + 1 \cdot 3 & 10 \\ 4 \cdot 2 + 5 \cdot 1 + 6 \cdot 3 & 31 \\ 3 \cdot 2 + 1 \cdot 1 + 5 \cdot 3 & 22 \end{array} \right) = \left(\begin{array}{c|c} 10 & 10 \\ 31 & 31 \\ 22 & 22 \end{array} \right)$$

1.4 Метод Гаусса с выбором главного элемента

На практике, часто может оказаться, что система (2) имеет единственное решение, хотя какой-либо из угловых миноров матрицы \mathbf{A} равен нулю. Кроме того, заранее обычно неизвестно, все ли угловые миноры матрицы \mathbf{A} отличны от нуля. В этих случаях обычный метод Гаусса может оказаться *непригодным*. Избежать указанных трудностей позволяет метод Гаусса с выбором главного элемента.

Основная идея метода состоит в том, чтобы на очередном шаге исключать не следующее по номеру неизвестное, а то неизвестное, коэффициент при котором является *наибольшим по модулю*. Таким образом, в качестве ведущего элемента здесь выбирается *главный*, т.е. наибольший по модулю элемент. Поэтому, если $\det \mathbf{A} \neq 0$, то в процессе вычислений не будет происходить деление на нуль.

На практике чаще всего применяется и метод Гаусса с выбором главного элемента по всей матрице, когда в качестве ведущего выбирается максимальный по модулю элемент *среди всех элементов* матрицы системы.

1.5 Численное решение системы линейных алгебраических уравнений методом Гаусса с выбором главного элемента по всей матрице

Рассмотрим на примере решение системы линейных уравнений методом Гаусса с выбором главного элемента по всей матрице системы:

$$\begin{cases} 2x_1 + 3x_2 + x_3 = 10 \\ 4x_1 + 5x_2 + 6x_3 = 31 \\ 3x_1 + x_2 + 5x_3 = 22 \end{cases} \iff \left(\begin{array}{ccc|c} 2 & 3 & 1 & 10 \\ 4 & 5 & 6 & 31 \\ 3 & 1 & 5 & 22 \end{array} \right)$$

Прямой ход метода Гаусса с выбором главного элемента.

- 1) Выбираем максимальный по модулю элемент в матрице (главный элемент) во второй строке, третьем столбце (выделен цветом):

$$\left(\begin{array}{ccc|c} 2 & 3 & 1 & 10 \\ 4 & 5 & 6 & 31 \\ 3 & 1 & 5 & 22 \end{array} \right) \rightarrow \left(\begin{array}{ccc|c} 2 & 3 & 1 & 10 \\ 4 & 5 & \mathbf{6} & 31 \\ 3 & 1 & 5 & 22 \end{array} \right)$$

Разделим каждую строку матрицы на значение элемента матрицы в столбце главного элемента, т.е. первую строку делим на 1, вторую строку на 6, а третью строку на 5:

$$\left(\begin{array}{ccc|c} 2 & 3 & 1 & 10 \\ 4 & 5 & \mathbf{6} & 31 \\ 3 & 1 & 5 & 22 \end{array} \right) \rightarrow \left(\begin{array}{ccc|c} 2 & 3 & \mathbf{1} & 10 \\ \frac{4}{6} & \frac{5}{6} & \mathbf{1} & \frac{31}{6} \\ \frac{3}{5} & \frac{1}{5} & \mathbf{1} & \frac{22}{5} \end{array} \right)$$

Вычитаем из первой и третьей строки вторую строку:

$$\left(\begin{array}{ccc|c} 2 & 3 & 1 & 10 \\ \frac{4}{6} & \frac{5}{6} & \mathbf{1} & \frac{31}{6} \\ \frac{3}{5} & \frac{1}{5} & 1 & \frac{22}{5} \end{array} \right) \rightarrow \left(\begin{array}{ccc|c} \frac{4}{3} & \frac{13}{6} & 0 & \frac{29}{6} \\ \frac{4}{6} & \frac{5}{6} & \mathbf{1} & \frac{31}{6} \\ -\frac{1}{15} & -\frac{19}{30} & 0 & -\frac{23}{30} \end{array} \right)$$

- 2) Исключаем из рассмотрения строку с текущим главным элементом (вторую строку) и выбираем новый главный элемент матрицы (первая строка, второй

столбец):

$$\left(\begin{array}{ccc|c} \frac{4}{3} & \frac{13}{6} & 0 & \frac{29}{6} \\ \frac{4}{6} & \frac{5}{6} & \boxed{1} & \frac{31}{6} \\ -\frac{1}{15} & -\frac{19}{30} & 0 & -\frac{23}{30} \end{array} \right) \rightarrow \left(\begin{array}{ccc|c} \frac{4}{3} & \boxed{\frac{13}{6}} & 0 & \frac{29}{6} \\ \frac{4}{6} & \frac{5}{6} & \boxed{1} & \frac{31}{6} \\ -\frac{1}{15} & -\frac{19}{30} & 0 & -\frac{23}{30} \end{array} \right)$$

Делим каждую строку матрицы на значение элемента матрицы в столбце главного элемента, т.е. первую строку делим на $\frac{13}{6}$, а третью строку на $-\frac{19}{30}$:

$$\left(\begin{array}{ccc|c} \frac{4}{3} & \boxed{\frac{13}{6}} & 0 & \frac{29}{6} \\ \frac{4}{6} & \frac{5}{6} & \boxed{1} & \frac{31}{6} \\ -\frac{1}{15} & -\frac{19}{30} & 0 & -\frac{23}{30} \end{array} \right) \rightarrow \left(\begin{array}{ccc|c} \frac{8}{13} & \boxed{1} & 0 & \frac{29}{13} \\ \frac{4}{6} & \frac{5}{6} & \boxed{1} & \frac{31}{6} \\ \frac{2}{19} & \boxed{1} & 0 & \frac{23}{19} \end{array} \right)$$

Вычитаем из третьей строки матрицы её первую строку:

$$\left(\begin{array}{ccc|c} \frac{8}{13} & \boxed{1} & 0 & \frac{29}{13} \\ \frac{4}{6} & \frac{5}{6} & \boxed{1} & \frac{31}{6} \\ \frac{2}{19} & 1 & 0 & \frac{23}{19} \end{array} \right) \rightarrow \left(\begin{array}{ccc|c} \frac{8}{13} & \boxed{1} & 0 & \frac{29}{13} \\ \frac{4}{6} & \frac{5}{6} & \boxed{1} & \frac{31}{6} \\ -\frac{126}{247} & 0 & 0 & -\frac{252}{247} \end{array} \right)$$

- 3) Исключаем из рассмотрения строку с текущим главным элементом (первую) и выбираем новый главный элемент матрицы (третья строка, первый столбец):

$$\left(\begin{array}{ccc|c} \frac{8}{13} & \boxed{1} & 0 & \frac{29}{13} \\ \frac{4}{6} & \frac{5}{6} & \boxed{1} & \frac{31}{6} \\ -\frac{126}{247} & 0 & 0 & -\frac{252}{247} \end{array} \right) \rightarrow \left(\begin{array}{ccc|c} \frac{8}{13} & \boxed{1} & 0 & \frac{29}{13} \\ \frac{4}{6} & \frac{5}{6} & \boxed{1} & \frac{31}{6} \\ \boxed{-\frac{126}{247}} & 0 & 0 & -\frac{252}{247} \end{array} \right)$$

Делим каждую строку матрицы на значение элемента матрицы в столбце главного элемента, т.е. третью строку делим на $-\frac{126}{247}$:

$$\left(\begin{array}{ccc|c} \frac{8}{13} & \boxed{1} & 0 & \frac{29}{13} \\ \frac{4}{6} & \frac{5}{6} & \boxed{1} & \frac{31}{6} \\ \boxed{-\frac{126}{247}} & 0 & 0 & -\frac{252}{247} \end{array} \right) \rightarrow \left(\begin{array}{ccc|c} \frac{8}{13} & \boxed{1} & 0 & \frac{29}{13} \\ \frac{4}{6} & \frac{5}{6} & \boxed{1} & \frac{31}{6} \\ \boxed{1} & 0 & 0 & 2 \end{array} \right)$$

Обратный ход метода Гаусса с выбором главного элемента.

Определим неизвестные из уравнений системы в обратном порядке сле-

дования номеров столбцов главных элементов, т.е. $x_1 \rightarrow x_2 \rightarrow x_3$:

1) Из третьего уравнения системы определим неизвестное x_1 :

$$x_1 = 2$$

2) Из первого уравнения системы определим неизвестное x_2 :

$$\frac{8}{13}x_1 + x_2 = \frac{29}{13} \rightarrow x_2 = -\frac{8}{13}x_1 + \frac{29}{13}$$

$$x_2 = -\frac{8}{13} \cdot 2 + \frac{29}{13} = 1$$

3) Из второго уравнения системы определим неизвестное x_3 :

$$\frac{4}{6}x_1 + \frac{5}{6}x_2 + x_3 = \frac{31}{6} \rightarrow x_3 = -\frac{4}{6}x_1 - \frac{5}{6}x_2 + \frac{31}{6}$$

$$x_3 = -\frac{4}{6} \cdot 2 - \frac{5}{6} \cdot 1 + \frac{31}{6} = 3$$

Таким образом, найдено решение системы линейных уравнений:

$$\mathbf{s} = \begin{pmatrix} 2 \\ 1 \\ 3 \end{pmatrix} \iff \mathbf{s} = (2, 1, 3)^T$$

2 Интерполирование функций

Задача интерполирования состоит в том, чтобы по известным значениям функции $f(x)$ в отдельных точках отрезка восстановить её значения в остальных точках этого отрезка. Такая постановка задачи допускает множество решений.

Например, задача интерполирования возникает, в том случае, когда известны результаты измерения $y_i = f(x_i)$ некоторой физической величины f в ограниченном количестве точек x_i ($i = 0, 1, \dots, n$), а требуется оценить значения этой величины в других точках.

Интерполирование используется также, когда вычисление значений $f(x)$ трудоемко, например, значение искомой функции может быть определено как решение сложной задачи, в которой x играет роль параметра. При этом можно вычислить небольшую таблицу значений функции, но прямое нахождение функции при большом числе значений аргумента практически затруднительно или нецелесообразно.

2.1 Линейная интерполяция функции

При *линейной интерполяции* функция $f(x)$ на отрезке $x \in [a, b]$ заменяется обобщенным интерполяционным полиномом $p_n(x)$, который построен в виде линейной комбинации $(n + 1)$ аналитических функций $\{\phi_i(x)\}$

$$p_n(x) = c_0 \cdot \phi_0(x) + c_1 \cdot \phi_1(x) + \dots + c_n \cdot \phi_n(x) = \sum_{i=0}^n c_i \cdot \phi_i(x), \quad (5)$$

таким образом, чтобы значения полинома $p_n(x)$ в определённых точках отрезка $\{x_0, x_1, \dots, x_n\}$ (узлах сетки) совпадают со значениями функции в этих точках $\{y_0, y_1, \dots, y_n\}$ (условия сопряжения):

$$\begin{cases} p_n(x_0) = y_0 \\ p_n(x_1) = y_1 \\ \dots\dots\dots \\ p_n(x_n) = y_n \end{cases}, \quad \Leftrightarrow \quad \begin{cases} \sum_{i=0}^n c_i \cdot \phi_i(x_0) = y_0 \\ \sum_{i=0}^n c_i \cdot \phi_i(x_1) = y_1 \\ \dots\dots\dots \\ \sum_{i=0}^n c_i \cdot \phi_i(x_n) = y_n \end{cases}. \quad (6)$$

Из условий (6), накладываемых на интерполяционный полином, форму-

лируется система линейных уравнений относительно неизвестных коэффициентов полинома c_0, c_1, \dots, c_n :

$$\mathbf{A} \cdot \mathbf{c} = \mathbf{y}, \quad (7)$$

где \mathbf{A} – квадратная матрица $(n + 1) \times (n + 1)$, \mathbf{c} и \mathbf{y} – вектор неизвестных коэффициентов полинома $p_n(x)$ и вектор значений функции $f(x)$ в заданных точках $\{x_i\}$:

$$\mathbf{A} = \begin{pmatrix} \phi_0(x_0) & \phi_1(x_0) & \cdots & \phi_n(x_0) \\ \phi_0(x_1) & \phi_1(x_1) & \cdots & \phi_n(x_1) \\ \vdots & \vdots & \ddots & \vdots \\ \phi_0(x_n) & \phi_1(x_n) & \cdots & \phi_n(x_n) \end{pmatrix}, \quad \mathbf{c} = \begin{pmatrix} c_0 \\ c_1 \\ \vdots \\ c_n \end{pmatrix}, \quad \mathbf{y} = \begin{pmatrix} y_0 \\ y_1 \\ \vdots \\ y_n \end{pmatrix}.$$

Если среди узлов интерполяции $\{x_i\}$ нет совпадающих ($x_i \neq x_j$ для всех $i, j = 0, 1, \dots, n$) и определитель системы отличен от нуля $\det \mathbf{A} \neq 0$, то задача интерполяции имеет единственное решение, а система функций $\{\phi_i(x)\}$ называется чебышевской. Поэтому при линейной интерполяции необходимо строить обобщенный полином $p_n(x)$ на основе *чебышевской системы функций*.

Таким образом, для определения коэффициентов интерполяционного полинома (5) необходимо найти решение системы линейных уравнений (7), любыми аналитическими, приближенными или численными методами, например, методом Гаусса.

Интерполирование не всегда дает удовлетворительное решение задачи о приближении функции *с заданной точностью* на данном промежутке, так как совпадение функции $f(x)$ с полиномом $p(x)$ в точках x_i и x_{i+1} не гарантирует малость величины $|f(x) - p(x)|$ на отрезке $[x_i, x_{i+1}]$.

2.2 Интерполяция алгебраическими полиномами

Задача интерполяции алгебраическими полиномами сводится к построению полинома степени n по чебышевской системе алгебраических функций $\{1, x, x^2, \dots, x^n\}$:

$$p_n(x) = c_0 + c_1 \cdot x + c_2 \cdot x^2 + \dots + c_n \cdot x^n = \sum_{i=0}^n c_i \cdot x^i. \quad (8)$$

Определитель системы (7) представляет собой определителем Вандермонда, который отличен от нуля $\det \mathbf{A} \neq 0$, если среди точек $\{x_i\}$ нет совпадающих, т.е. $x_i \neq x_j$ для всех $i, j = 0, 1, \dots, n$:

$$\det \mathbf{A} = \begin{vmatrix} 1 & x_0 & \cdots & x_0^n \\ 1 & x_1 & \cdots & x_1^n \\ \vdots & \vdots & \ddots & \vdots \\ 1 & x_n & \cdots & x_n^n \end{vmatrix}.$$

Выражение для коэффициентов алгебраического полинома и вид самого полинома (8) можно записать различными способами. Наиболее распространенная запись интерполяционного многочлена в форме Лагранжа и в форме Ньютона.

2.3 Интерполяционный полином в форме Лагранжа

Интерполяционная формула Лагранжа позволяет представить многочлен $L_n(x)$ в виде линейной комбинации значений функции $y(x)$ в узлах интерполирования $\{x_i\}$:

$$L_n(x) = \lambda_0(x) \cdot y_0 + \lambda_1(x) \cdot y_1 + \cdots + \lambda_n(x) \cdot y_n = \sum_{i=0}^n \lambda_i(x) \cdot y_i \quad (9)$$

где $\lambda_0(x), \lambda_1(x), \dots, \lambda_n(x)$ – произвольные неизвестные функции.

Для определения неизвестных функций $\lambda_i(x)$ из условий интерполирования следует:

[illegible]

Эта система уравнений имеет решение если выполняются условия:

$$\lambda_i(x_j) = \begin{cases} 1, & x_j = x_i \\ 0, & x_j \neq x_i \end{cases}$$

Коэффициенты $\lambda_i(x)$ можно искать в виде многочленов степени n :

[illegible]

Определим неизвестные $\alpha_0, \alpha_1, \dots, \alpha_n$ из условия для коэффициентов $\lambda_i(x)$:

$$\left\{ \begin{array}{l} 1 = \alpha_0 \cdot (x_0 - x_1) \cdot (x_0 - x_2) \cdot (x_0 - x_2) \cdot \dots \cdot (x_0 - x_n) \\ 1 = \alpha_1 \cdot (x_1 - x_0) \cdot (x_1 - x_2) \cdot (x_1 - x_3) \cdot \dots \cdot (x_1 - x_n) \\ \vdots \\ 1 = \alpha_n \cdot (x_n - x_0) \cdot (x_n - x_1) \cdot (x_n - x_2) \cdot \dots \cdot (x_n - x_{n-1}) \end{array} \right.$$

Таким образом, коэффициенты $\lambda_i(x)$ интерполяционного многочлена Лагранжа находятся из соотношений:

$$\left\{ \begin{array}{l} \lambda_0(x) = \frac{(x - x_1) \cdot (x - x_2) \cdots (x - x_n)}{(x_0 - x_1) \cdot (x_0 - x_2) \cdots (x_0 - x_n)} \\ \lambda_1(x) = \frac{(x - x_0) \cdot (x - x_2) \cdots (x - x_n)}{(x_1 - x_0) \cdot (x_1 - x_2) \cdots (x_1 - x_n)} \\ \dots\dots\dots \\ \lambda_n(x) = \frac{(x - x_0) \cdot (x - x_1) \cdots (x - x_{n-1})}{(x_n - x_0) \cdot (x_n - x_1) \cdots (x_n - x_{n-1})} \end{array} \right.,$$

или в более компактной форме:

$$\lambda_i(x) = \frac{\prod_{j \neq i}^n (x - x_j)}{\prod_{j \neq i}^n (x_i - x_j)}$$

Итак, интерполяционный многочлен Лагранжа (9) имеет вид:

$$L_n(x) = \sum_{i=0}^n \frac{\prod_{j \neq i}^n (x - x_j)}{\prod_{j \neq i}^n (x_i - x_j)} \cdot y_i \quad (10)$$

2.4 Интерполяция функции заданной таблично обобщенным полиномом

Известно множество данных (узлов интерполяции) $\{x_i\}$, в которых определены значения функции $y_i = f(x_i)$:

Таблица 1 – Таблично заданная функциональная зависимость

| i | 0 | 1 | 2 | 3 |
|-------|-------|-------|------|------|
| x_i | -0.76 | -0.09 | 0.22 | 0.55 |
| y_i | 0.08 | 1.84 | 0.40 | 0.96 |

Построим обобщенный интерполяционный полином $p_3(x)$ для таблично заданной функции исходя из чебышевской системы функций $\{1, x, e^{-x}, e^x\}$:

$$p_4(x) = c_0 + c_1 \cdot x + c_2 \cdot e^{-x} + c_3 \cdot e^x$$

- 1) Составим расширенную матрица системы уравнений (7) для определения коэффициентов полинома $(c_0, c_1, c_2, c_3)^T$:

$$\left(\begin{array}{cccc|c} \phi_0(x_0) & \phi_1(x_0) & \phi_2(x_0) & \phi_3(x_0) & y_0 \\ \phi_0(x_1) & \phi_1(x_1) & \phi_2(x_1) & \phi_3(x_1) & y_1 \\ \phi_0(x_2) & \phi_1(x_2) & \phi_2(x_2) & \phi_3(x_2) & y_2 \\ \phi_0(x_3) & \phi_1(x_3) & \phi_2(x_3) & \phi_3(x_3) & y_3 \end{array} \right), \quad \text{здесь} \quad \begin{cases} \phi_0(x) = 1 \\ \phi_1(x) = x \\ \phi_2(x) = e^{-x} \\ \phi_3(x) = e^x \end{cases}.$$

- 2) Воспользуемся данными таблицы 1 и заполним числовыми значения эле-

менты расширенной матрицы:

$$\left(\begin{array}{cccc|c} 1 & -0.76 & e^{0.76} & e^{-0.76} & 0.08 \\ 1 & -0.09 & e^{0.09} & e^{-0.09} & 1.84 \\ 1 & 0.22 & e^{-0.22} & e^{0.22} & 0.40 \\ 1 & 0.55 & e^{-0.55} & e^{0.55} & 0.96 \end{array} \right) \Leftrightarrow \left(\begin{array}{cccc|c} 1 & -0.76 & 2.138 & 0.468 & 0.08 \\ 1 & -0.09 & 1.094 & 0.914 & 1.84 \\ 1 & 0.22 & 0.803 & 1.246 & 0.40 \\ 1 & 0.55 & 0.577 & 1.733 & 0.96 \end{array} \right)$$

- 3) Решение системы линейных уравнений найдем методом Гаусса с выбором главного элемента в расширенной матрице (выделен цветом):

$$\mathbf{c} = (-0.393, -81.472, -37.288, 39.053)^T$$

Следовательно, обобщенный интерполяционный полином для функции заданной таблично можно записать в виде:

$$p_3(x) = -0.393 - 81.472 \cdot x - 37.288 \cdot e^{-x} + 39.053 \cdot e^x$$

- 4) В таблице 2 представлены данные расчета коэффициентов обобщенного интерполяционного полинома c_i , значений этого полинома в узлах сетки $p_3(x_i)$ и абсолютная погрешность интерполяции $\varepsilon_i = y_i - p_3(x_i)$.

Таблица 2 – Коэффициенты обобщенного интерполяционного полинома c_i , значения этого полинома в узлах сетки $p_3(x_i)$ и абсолютная погрешность интерполяции ε_i

| i | 0 | 1 | 2 | 3 |
|-----------------|--------|---------|---------|--------|
| x_i | -0.76 | -0.09 | 0.22 | 0.55 |
| y_i | 0.08 | 1.84 | 0.4 | 0.96 |
| c_i | -0.393 | -81.472 | -37.288 | 39.053 |
| $p_3(x_i)$ | 0.057 | 1.832 | 0.422 | 0.973 |
| ε_i | 0.023 | 0.008 | -0.022 | -0.013 |

- 5) На рисунке 1 представлена диаграмма рассеяния (разброса) данных функции

заданной таблично $y_i = f(x_i)$ (маркеры) и результаты вычислений обобщенного интерполяционного полинома $p_3(x)$ (сплошная линия).

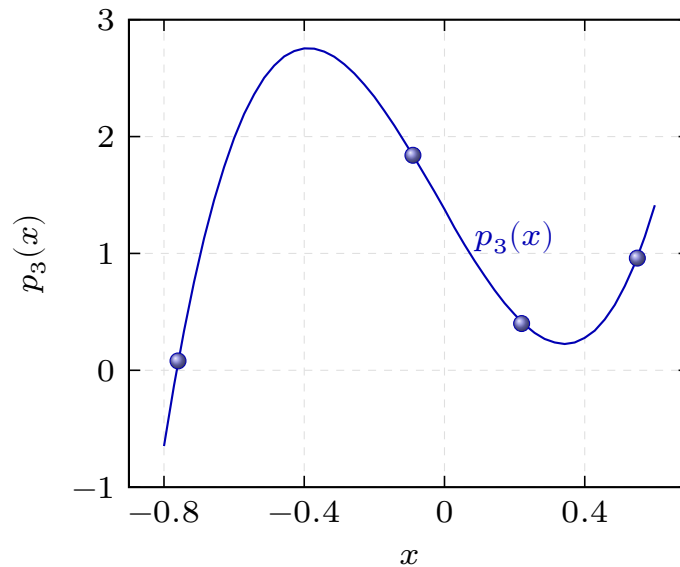


Рисунок 1 – График таблично заданной функции $y_i = f(x_i)$ (маркеры) и обобщенного интерполяционного полинома $p_3(x)$ (сплошная линия)

Таблица 3 – Рассчётные значения обобщенного интерполяционного полинома в узлах сетки $p_3(x_i)$

| i | 0 | 1 | 2 | ... | ... | ... | 47 | 48 | 49 |
|-------|--------|--------|--------|-----|-----|-----|-------|-------|-------|
| x_i | -0.800 | -0.771 | -0.743 | ... | ... | ... | 0.543 | 0.571 | 0.600 |
| y_i | -0.654 | -0.128 | 0.330 | ... | ... | ... | 0.920 | 1.145 | 1.419 |

2.5 Интерполяция функции заданной таблично алгебраическими полиномами в форме Лагранжа

Построим интерполяционный полином в форме Лагранжа $L_3(x)$ на основе данных об узлах интерполяции $\{x_i\}$ и значений функции в этих узлах $\{y_i\}$:

$$L_3(x) = \sum_{i=0}^3 \frac{\prod_{j \neq i}^3 (x - x_j)}{\prod_{j \neq i}^3 (x_i - x_j)} \cdot y_i$$

1) Представим полином Лагранжа в развернутом виде:

$$L_3(x) = \frac{(x - x_1) \cdot (x - x_2) \cdot (x - x_3)}{(x_0 - x_1) \cdot (x_0 - x_2) \cdot (x_0 - x_3)} \cdot y_0 +$$

$$\frac{(x - x_0) \cdot (x - x_2) \cdot (x - x_3)}{(x_1 - x_0) \cdot (x_1 - x_2) \cdot (x_1 - x_3)} \cdot y_1 +$$

$$\frac{(x - x_0) \cdot (x - x_1) \cdot (x - x_3)}{(x_2 - x_0) \cdot (x_2 - x_1) \cdot (x_2 - x_3)} \cdot y_2 +$$

$$\frac{(x - x_0) \cdot (x - x_1) \cdot (x - x_2)}{(x_3 - x_0) \cdot (x_3 - x_1) \cdot (x_3 - x_2)} \cdot y_3$$

2) Воспользуемся численными данными об узлах интерполяции $\{x_i\}$ и известными значениями интерпретируемой функции в этих узлах $\{y_i\}$ (таблица 1):

$$L_3(x) = \frac{(x - (-0.09)) \cdot (x - 0.22) \cdot (x - 0.55)}{(-0.76 - (-0.09)) \cdot (-0.76 - 0.22) \cdot (-0.76 - 0.55)} \cdot 0.08 +$$

$$\frac{(x - (-0.76)) \cdot (x - 0.22) \cdot (x - 0.55)}{(-0.09 - (-0.76)) \cdot (-0.09 - 0.22) \cdot (-0.09 - 0.55)} \cdot 1.84 +$$

$$\frac{(x - (-0.76)) \cdot (x - (-0.09)) \cdot (x - 0.55)}{(0.22 - (-0.76)) \cdot (0.22 - (-0.09)) \cdot (0.22 - 0.55)} \cdot 0.40 +$$

$$\frac{(x - (-0.76)) \cdot (x - (-0.09)) \cdot (x - 0.22)}{(0.55 - (-0.76)) \cdot (0.55 - (-0.09)) \cdot (0.55 - 0.22)} \cdot 0.96$$

3) Проведем необходимые арифметические действия:

$$L_3(x) = \frac{(x + 0.09) \cdot (x - 0.22) \cdot (x - 0.55)}{(-0.67) \cdot (-0.98) \cdot (-1.31)} \cdot 0.08 +$$

$$\frac{(x + 0.76) \cdot (x - 0.22) \cdot (x - 0.55)}{(0.67) \cdot (-0.31) \cdot (-0.64)} \cdot 1.84 +$$

$$\frac{(x + 0.76) \cdot (x + 0.09) \cdot (x - 0.55)}{(0.98) \cdot (0.31) \cdot (-0.33)} \cdot 0.40 +$$

$$\frac{(x + 0.76) \cdot (x + 0.09) \cdot (x - 0.22)}{(1.31) \cdot (0.64) \cdot (0.33)} \cdot 0.96$$

или

$$L_3(x) = \frac{(x + 0.09) \cdot (x - 0.22) \cdot (x - 0.55)}{-0.86} \cdot 0.08 +$$

$$\frac{(x + 0.76) \cdot (x - 0.22) \cdot (x - 0.55)}{0.13} \cdot 1.84 +$$

$$\frac{(x + 0.76) \cdot (x + 0.09) \cdot (x - 0.55)}{-0.10} \cdot 0.40 +$$

$$\frac{(x + 0.76) \cdot (x + 0.09) \cdot (x - 0.22)}{0.28} \cdot 0.96$$

Продолжая делать упрощения окончательно получим:

$$L_3(x) = (x + 0.09) \cdot (x - 0.22) \cdot (x - 0.55) \cdot (-0.09) +$$

$$(x + 0.76) \cdot (x - 0.22) \cdot (x - 0.55) \cdot 13.84 +$$

$$(x + 0.76) \cdot (x + 0.09) \cdot (x - 0.55) \cdot (-3.99) +$$

$$(x + 0.76) \cdot (x + 0.09) \cdot (x - 0.22) \cdot 3.47$$

- 4) Запишем выражение для интерполяционный полином Лагранжа в каноническом виде:

$$L_3(x) = 1.37 - 5.248 \cdot x + 0.912 \cdot x^2 + 13.23 \cdot x^3$$

- 5) В таблице 4 представлены данные расчета коэффициентов интерполяционного полинома Лагранжа c_i , значений этого полинома в узлах сетки $L_3(x_i)$ и абсолютная погрешность интерполяции $\varepsilon_i = y_i - L_3(x_i)$.

Таблица 4 – Коэффициенты интерполяционного полинома Лагранжа c_i , значения этого полинома в узлах сетки $L_3(x_i)$ и абсолютная погрешность интерполяции ε_i

| i | x_i | y_i | c_i | $L_3(x_i)$ | ε_i |
|-----|-------|-------|--------|------------|-----------------|
| 0 | -0.76 | 0.08 | 1.37 | 0.078 | 0.002 |
| 1 | -0.09 | 1.84 | -5.248 | 1.840 | 0.000 |
| 2 | 0.22 | 0.40 | 0.912 | 0.400 | 0.000 |
| 3 | 0.55 | 0.96 | 13.23 | 0.961 | -0.001 |

- 6) На рисунке (2) представлена диаграмму рассеяния (разброса) данных функции заданной таблично $y_i = f(x_i)$ (маркеры) и результаты вычислений интерполяционного полинома Лагранжа $L_3(x)$ (сплошная линия).

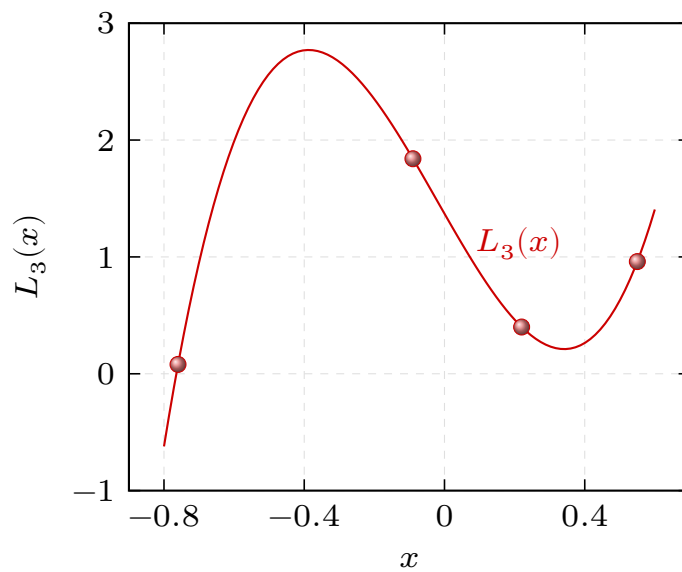


Рисунок 2 – График таблично заданной функции $y_i = f(x_i)$ (маркеры) и интерполяционного полинома Лагранжа $L_3(x)$ (сплошная линия)

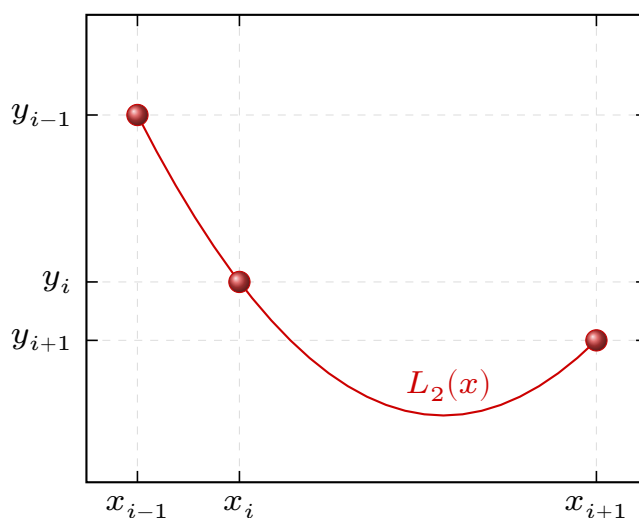
3 Численное дифференцирование

Задача численного дифференцирования состоит в приближенном вычислении производных $y'(x)$ по заданным в конечном числе точек $\{x_i\}$ значениям этой функции $\{y_i\}$.

Численное дифференцирование применяется, если функцию $y(x)$ трудно или невозможно продифференцировать аналитически, например, если функция является таблично заданной, а также при решении дифференциальных уравнений разностными методами.

Многие формулы численного дифференцирования можно получить, используя интерполяционные формулы. Для этого достаточно заменить функцию $y(x)$ интерполяционным полиномом, например, алгебраическим полиномом в форме Лагранжа $L_n(x)$, и вычислить производные этого многочлена, используя его явное представление.

Рассмотрим произвольную сетку $\{x_i\}$ и проведем интерполирование функции $y(x)$ в узлах сетки $x_{i-1} < x_i < x_{i+1}$ полиномом Лагранжа второго порядка, приближенно полагая $y(x) \approx L_2(x)$ для $x \in [x_{i-1}, x_{i+1}]$:



$$\begin{aligned}
L_2(x) &= \frac{(x - x_i) \cdot (x - x_{i+1})}{(x_{i-1} - x_i) \cdot (x_{i-1} - x_{i+1})} \cdot y_{i-1} + \\
&+ \frac{(x - x_{i-1}) \cdot (x - x_{i+1})}{(x_i - x_{i-1}) \cdot (x_i - x_{i+1})} \cdot y_i + \\
&+ \frac{(x - x_{i-1}) \cdot (x - x_i)}{(x_{i+1} - x_{i-1}) \cdot (x_{i+1} - x_i)} \cdot y_{i+1}
\end{aligned}$$

где $y_{i-1} = y(x_{i-1})$, $y_i = y(x_i)$, $y_{i+1} = y(x_{i+1})$ – значение функции $y(x)$ в узлах сетки.

Первая производная многочлена Лагранжа $L_2(x)$:

$$\begin{aligned}
L'_2(x) &= \frac{2x - x_i - x_{i+1}}{(x_{i-1} - x_i) \cdot (x_{i-1} - x_{i+1})} \cdot y_{i-1} + \\
&+ \frac{2x - x_{i-1} - x_{i+1}}{(x_i - x_{i-1}) \cdot (x_i - x_{i+1})} \cdot y_i + \\
&+ \frac{2x - x_{i-1} - x_i}{(x_{i+1} - x_{i-1}) \cdot (x_{i+1} - x_i)} \cdot y_{i+1}
\end{aligned}$$

Это выражение можно принять за приближенное значение первой производной $y'(x)$ в любой точке отрезка $[x_{i-1}, x_{i+1}]$. Например, в точке $x = x_i$ первая производная от функции $y(x)$ приближенно равна:

$$\begin{aligned}
y'(x_i) \approx L'_2(x_i) &= \frac{x_i - x_{i+1}}{(x_{i-1} - x_i) \cdot (x_{i-1} - x_{i+1})} \cdot y_{i-1} + \\
&+ \frac{(x_i - x_{i-1}) + (x_i - x_{i+1})}{(x_i - x_{i-1}) \cdot (x_i - x_{i+1})} \cdot y_i + \\
&+ \frac{x_i - x_{i-1}}{(x_{i+1} - x_{i-1}) \cdot (x_{i+1} - x_i)} \cdot y_{i+1}
\end{aligned}$$

Вторую производную полинома Лагранжа можно принять за приближенное значение второй производной от функции $y(x)$ в любой точке отрезка

$[x_{i-1}, x_{i+1}]$:

$$\begin{aligned} y''(x) \approx L_2''(x) = & \frac{2}{(x_{i-1} - x_i) \cdot (x_{i-1} - x_{i+1})} \cdot y_{i-1} + \\ & + \frac{2}{(x_i - x_{i-1}) \cdot (x_i - x_{i+1})} \cdot y_i + \\ & + \frac{2}{(x_{i+1} - x_{i-1}) \cdot (x_{i+1} - x_i)} \cdot y_{i+1} \end{aligned}$$

На *равномерной сетке* $\{x_i\}$, расстояние между соседними узлами которой одинаково, выражения для первой и второй производной в точке $x = x_i$ упрощаются:

$$y'(x_i) \approx \frac{y_{i+1} - y_{i-1}}{2h}, \quad y''(x_i) \approx \frac{y_{i-1} - 2y_i + y_{i+1}}{h^2},$$

где $h = (x_i - x_{i-1}) = (x_{i+1} - x_i)$ – шаг сетки.

Для приближенного вычисления производных более высоких порядков $y^{(n)}(x)$ уже недостаточно полинома Лагранжа второго порядка $L_2(x)$. Поэтому необходимо использовать полиномы более высокого порядка, что приводит к увеличению числа узлов аппроксимации.

Следует отметить, что порядок погрешности аппроксимации производных от функции $y(x)$ зависит как от порядка интерполяционного полинома, так и от расположения узлов сетки $\{x_i\}$.

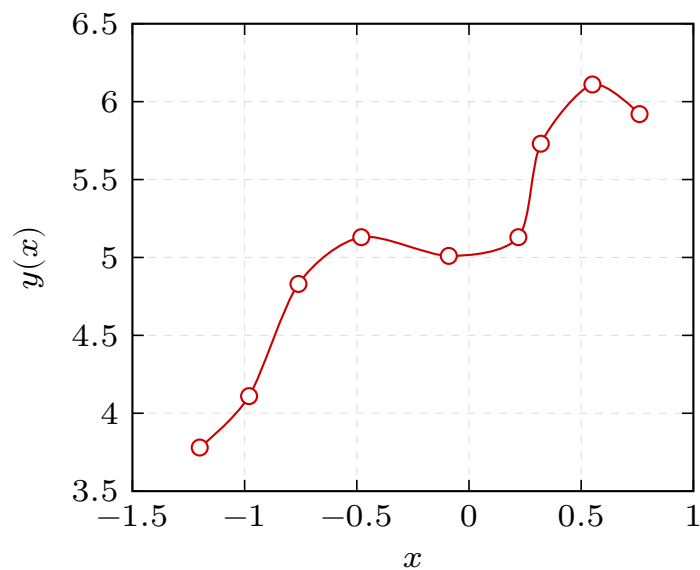
3.1 Численного дифференцирование функции заданной таблично

Известно множество данных (узлов сетки) $\{x_i\}$ в которых определены значения функции $\{f(x_i)\}$:

Таблица 5 – Таблично заданная функциональная зависимость $y_i = f(x_i)$

| i | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 |
|-------|------|-------|-------|-------|-------|------|------|------|------|
| x_i | −1.2 | −0.98 | −0.76 | −0.48 | −0.09 | 0.22 | 0.32 | 0.55 | 0.76 |
| y_i | 3.78 | 4.11 | 4.83 | 5.13 | 5.01 | 5.13 | 5.73 | 6.11 | 5.92 |

1) Построим график функции $y(x)$, используя данные таблицы 5.

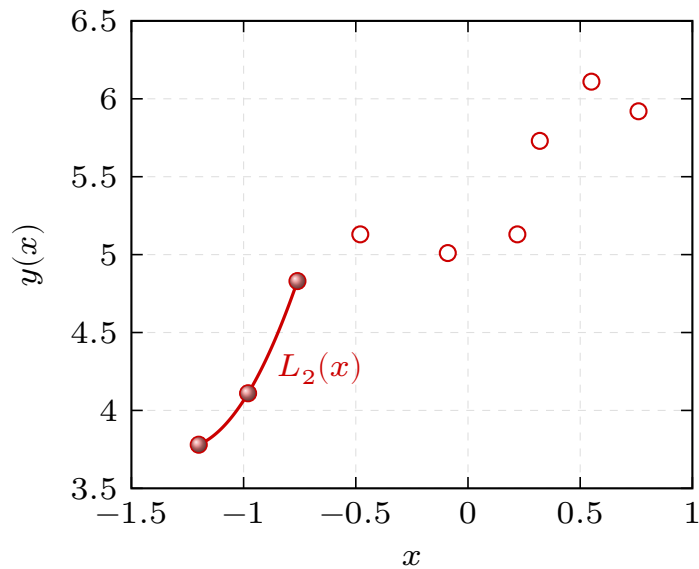


2) Аппроксимируем функцию $y(x)$ в узлах $\{x_0, x_1, x_2\}$ полиномом Лагранжа второго порядка $L_2(x)$, используя данные таблицы 5:

$$\begin{aligned}
 L_2(x) = & \frac{(x - (-0.98))(x - (-0.76))}{(-1.20 - (-0.98))(-1.20 - (-0.76))} \cdot 3.78 + \\
 & + \frac{(x - (-1.20))(x - (-0.76))}{(-0.98 - (-1.20))(-0.98 - (-0.76))} \cdot 4.11 + \\
 & + \frac{(x - (-1.20))(x - (-0.98))}{(-0.76 - (-1.20))(-0.76 - (-0.98))} \cdot 4.83
 \end{aligned}$$

Проводя элементарные алгебраические преобразования полином Лагранжа в пределах отрезка $[x_0, x_2]$ имеет вид:

$$L_2(x) = 4.028925620 \cdot x^2 + 10.28305785 \cdot x + 10.31801653$$



Определим первую и вторую производную функции $y(x)$ в точке $x_1 = -0.98$:

$$y'_1 = y'(x_1) = y'(-0.98) \approx L'_2(-0.98) = 2.386363635$$

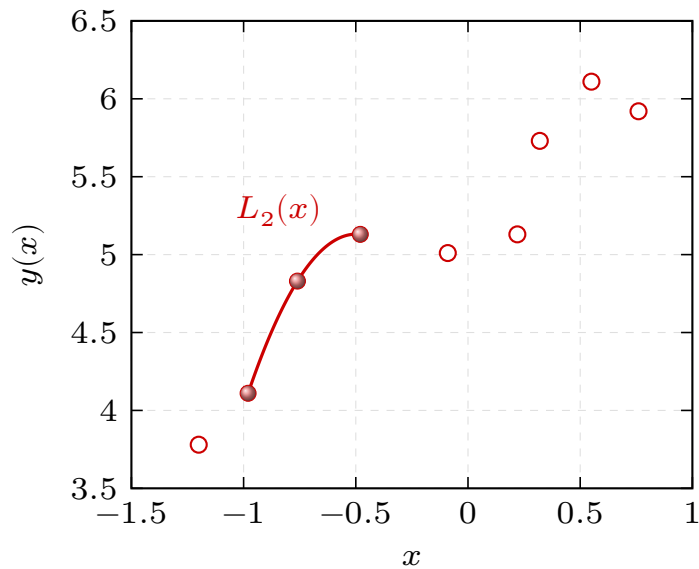
$$y''_1 = y''(x_1) = y''(-0.98) \approx L''_2(-0.98) = 8.057851240$$

3) Аппроксимация функции $y(x)$ в узлах $\{x_1, x_2, x_3\}$ полиномом Лагранжа второго порядка $L_2(x)$, используя данные таблицы 5:

$$\begin{aligned} L_2(x) = & \frac{(x - (-0.76))(x - (-0.48))}{(-0.98 - (-0.76))(-0.98 - (-0.48))} \cdot 4.11 + \\ & + \frac{(x - (-0.98))(x - (-0.48))}{(-0.76 - (-0.98))(-0.76 - (-0.48))} \cdot 4.83 + \\ & + \frac{(x - (-0.98))(x - (-0.76))}{(-0.48 - (-0.98))(-0.48 - (-0.76))} \cdot 5.13 \end{aligned}$$

Проводя элементарные алгебраические преобразования полином Лагранжа в пределах отрезка $[x_1, x_3]$ имеет вид:

$$L_2(x) = -4.402597390 \cdot x^2 - 4.387792189 \cdot x + 4.038218187$$



Определим первую и вторую производную функции $y(x)$ в точке $x_2 = -0.76$:

$$y'_2 = y'(x_2) = y'(-0.76) \approx L'_2(-0.76) = 2.304155844$$

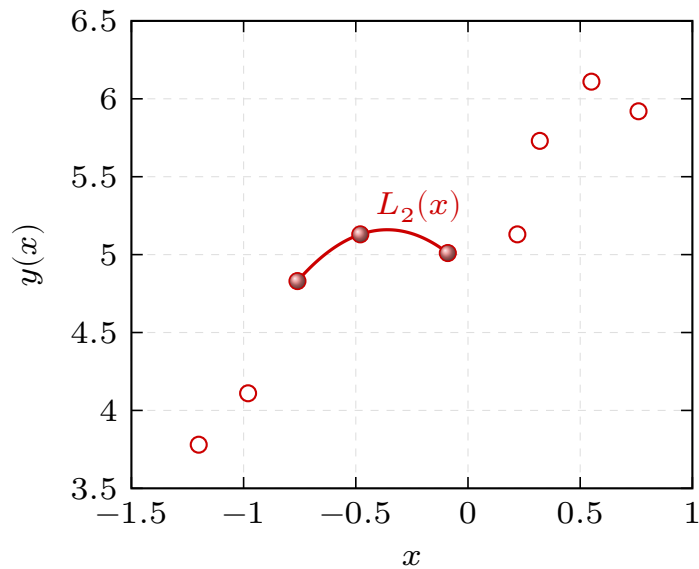
$$y''_2 = y''(x_2) = y''(-0.76) \approx L''_2(-0.76) = -8.805194780$$

- 4) Аппроксимация функции $y(x)$ в узлах $\{x_2, x_3, x_4\}$ полиномом Лагранжа второго порядка $L_2(x)$, используя данные таблицы 5:

$$\begin{aligned} L_2(x) = & \frac{(x - (-0.48))(x - (-0.09))}{(-0.76 - (-0.48))(-0.76 - (-0.09))} \cdot 4.83 + \\ & + \frac{(x - (-0.76))(x - (-0.09))}{(-0.48 - (-0.76))(-0.48 - (-0.09))} \cdot 5.13 + \\ & + \frac{(x - (-0.76))(x - (-0.48))}{(-0.09 - (-0.76))(-0.09 - (-0.48))} \cdot 5.01 \end{aligned}$$

Проводя элементарные алгебраические преобразования полином Лагранжа в пределах отрезка $[x_2, x_4]$ имеет вид:

$$L_2(x) = -2.058389370 \cdot x^2 - 1.480974249 \cdot x + 4.893385272$$



Определим первую и вторую производную функции $y(x)$ в точке $x_3 = -0.48$:

$$y'_3 = y'(x_3) = y'(-0.48) \approx L'_2(-0.48) = 0.495079546$$

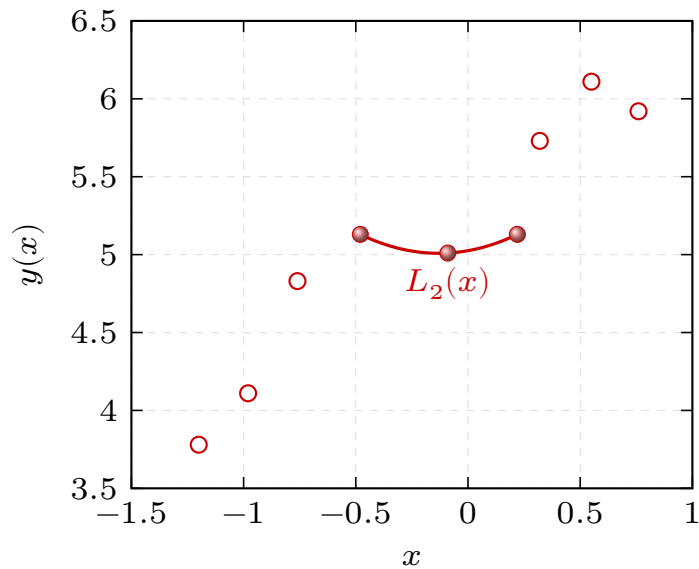
$$y''_3 = y''(x_3) = y''(-0.48) \approx L''_2(-0.48) = -4.116778740$$

5) Аппроксимация функции $y(x)$ в узлах $\{x_3, x_4, x_5\}$ полиномом Лагранжа второго порядка $L_2(x)$, используя данные таблицы 5:

$$\begin{aligned} L_2(x) = & \frac{(x - (-0.09))(x - 0.22)}{(-0.48 - (-0.09))(-0.48 - 0.22)} \cdot 5.13 + \\ & + \frac{(x - (-0.48))(x - 0.22)}{(-0.09 - (-0.48))(-0.09 - 0.22)} \cdot 5.01 + \\ & + \frac{(x - (-0.48))(x - (-0.09))}{(0.22 - (-0.48))(0.22 - (-0.09))} \cdot 5.13 \end{aligned}$$

Проводя элементарные алгебраические преобразования полином Лагранжа в пределах отрезка $[x_3, x_5]$ имеет вид:

$$L_2(x) = 0.9925558300 \cdot x^2 + 0.2580645177 \cdot x + 5.025186105$$



Определим первую и вторую производную функции $y(x)$ в точке $x_4 = -0.09$:

$$y'_4 = y'(x_4) = y'(-0.09) \approx L'_2(-0.09) = 0.0794044683$$

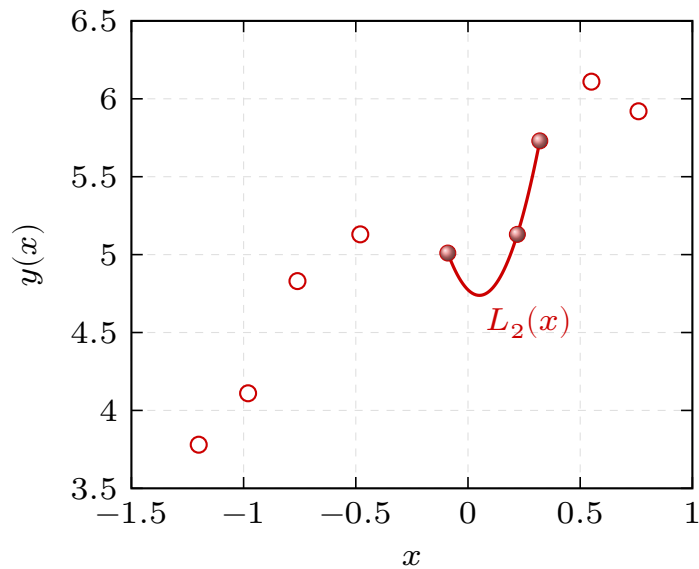
$$y''_4 = y''(x_4) = y''(-0.09) \approx L''_2(-0.09) = 1.985111660$$

- 6) Аппроксимация функции $y(x)$ в узлах $\{x_4, x_5, x_6\}$ полиномом Лагранжа второго порядка $L_2(x)$, используя данные таблицы 5:

$$\begin{aligned} L_2(x) &= \frac{(x - 0.22)(x - 0.32)}{(-0.09 - 0.22)(-0.09 - 0.32)} \cdot 5.01 + \\ &+ \frac{(x - (-0.09))(x - 0.32)}{(0.22 - (-0.09))(0.22 - 0.32)} \cdot 5.13 + \\ &+ \frac{(x - (-0.09))(x - 0.22)}{(0.32 - (-0.09))(0.32 - 0.22)} \cdot 5.73 \end{aligned}$$

Проводя элементарные алгебраические преобразования полином Лагранжа в пределах отрезка $[x_4, x_6]$ имеет вид:

$$L_2(x) = 13.69000778 \cdot x^2 - 1.392604236 \cdot x + 4.773776556$$



Определим первую и вторую производную функции $y(x)$ в точке $x_5 = 0.22$:

$$y'_5 = y'(x_5) = y'(0.22) \approx L'_2(0.22) = 4.630999187$$

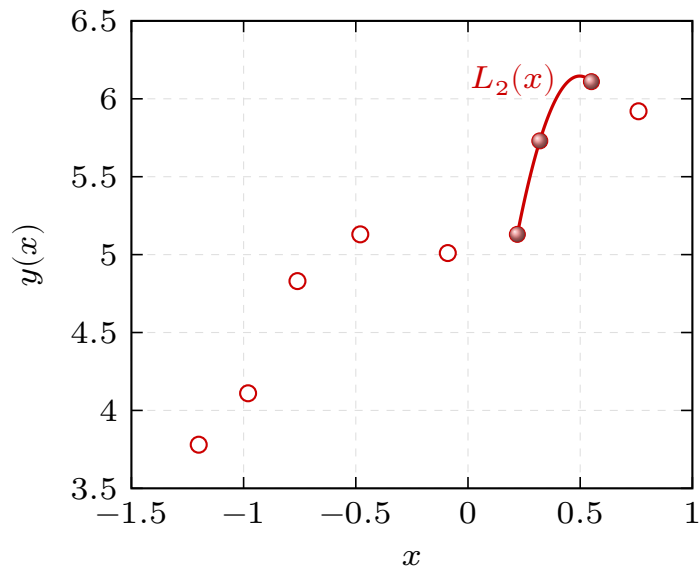
$$y''_5 = y''(x_5) = y''(0.22) \approx L''_2(0.22) = 27.38001556$$

7) Аппроксимация функции $y(x)$ в узлах $\{x_5, x_6, x_7\}$ полиномом Лагранжа второго порядка $L_2(x)$, используя данные таблицы 5:

$$\begin{aligned} L_2(x) &= \frac{(x - 0.32)(x - 0.55)}{(0.22 - 0.32)(0.22 - 0.55)} \cdot 5.13 + \\ &+ \frac{(x - 0.22)(x - 0.55)}{(0.32 - 0.22)(0.32 - 0.55)} \cdot 5.73 + \\ &+ \frac{(x - 0.22)(x - 0.32)}{(0.55 - 0.22)(0.55 - 0.32)} \cdot 6.11 \end{aligned}$$

Проводя элементарные алгебраические преобразования полином Лагранжа в пределах отрезка $[x_5, x_7]$ имеет вид:

$$L_2(x) = -13.17523062 \cdot x^2 + 13.11462456 \cdot x + 2.882463758$$



Определим первую и вторую производную функции $y(x)$ в точке $x_6 = 0.32$:

$$y'_6 = y'(x_6) = y'(0.32) \approx L'_2(0.32) = 4.682476963$$

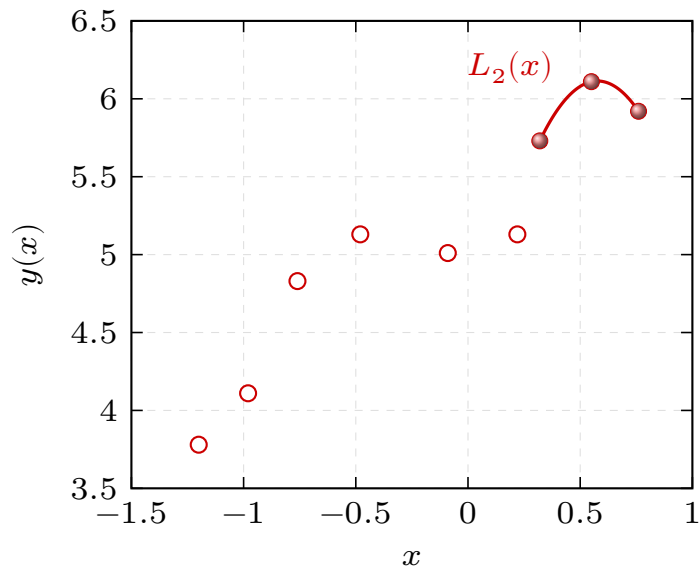
$$y''_6 = y''(x_6) = y''(0.32) \approx L''_2(0.32) = -26.35046124$$

8) Аппроксимация функции $y(x)$ в узлах $\{x_6, x_7, x_8\}$ полиномом Лагранжа второго порядка $L_2(x)$, используя данные таблицы 5:

$$\begin{aligned} L_2(x) &= \frac{(x - 0.55)(x - 0.76)}{(0.32 - 0.55)(0.32 - 0.76)} \cdot 5.73 + \\ &+ \frac{(x - 0.32)(x - 0.76)}{(0.55 - 0.32)(0.55 - 0.76)} \cdot 6.11 + \\ &+ \frac{(x - 0.32)(x - 0.55)}{(0.76 - 0.32)(0.76 - 0.55)} \cdot 5.92 \end{aligned}$$

Проводя элементарные алгебраические преобразования полином Лагранжа в пределах отрезка $[x_6, x_8]$ имеет вид:

$$L_2(x) = -5.811217790 \cdot x^2 + 6.707933391 \cdot x + 4.178530017$$



Определим первую и вторую производную функции $y(x)$ в точке $x_7 = 0.55$:

$$y'_7 = y'(x_7) = y'(0.55) \approx L'_2(0.55) = 0.315593822$$

$$y''_7 = y''(x_7) = y''(0.55) \approx L''_2(0.55) = -11.62243558$$

9) Таким образом, определены значения первой $y'(x)$ и второй $y''(x)$ производной функции $y(x)$ в каждом внутреннем узле сетки:

Таблица 6 – Значения первой и второй производных функции $y(x)$ во внутренних узлах сетки $\{x_i\}$

| i | 1 | 2 | 3 | 4 | 5 | 6 | 7 |
|---------|-------|-------|-------|-------|-------|--------|--------|
| x_i | -0.98 | -0.76 | -0.48 | -0.09 | 0.22 | 0.32 | 0.55 |
| y'_i | 2.39 | 2.30 | 0.50 | 0.08 | 4.63 | 4.68 | 0.32 |
| y''_i | 8.06 | -8.81 | -4.12 | 1.99 | 27.38 | -26.35 | -11.62 |

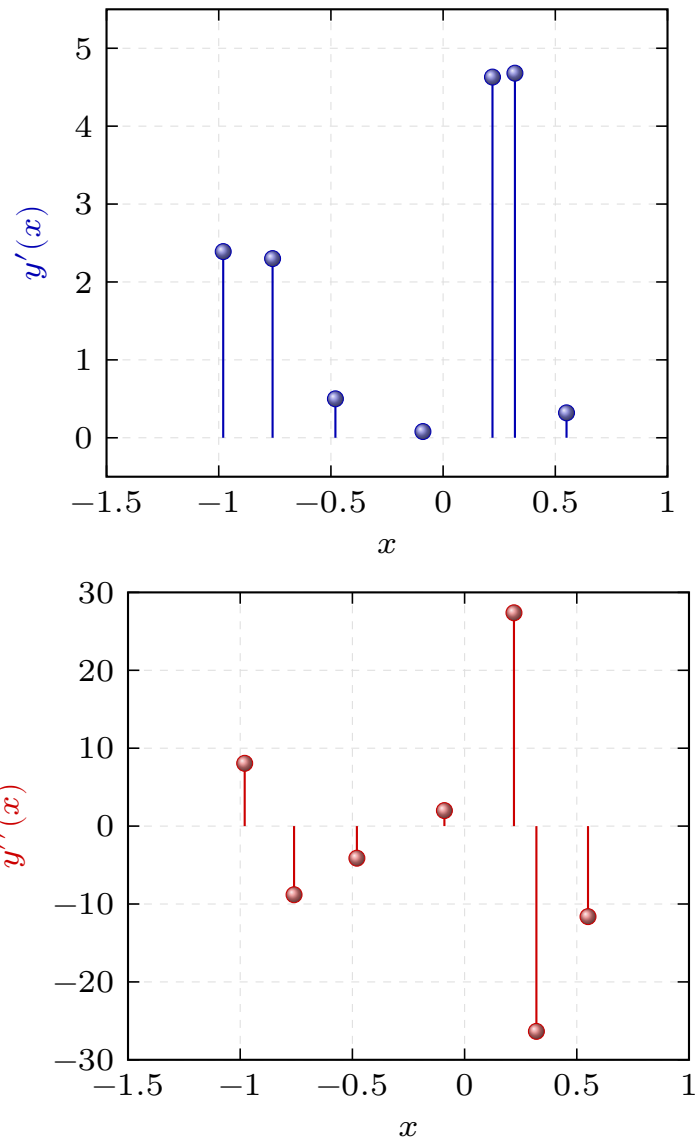


Рисунок 3 – Графики первой $y'(x)$ и второй $y''(x)$ производной от функции $y(x)$, заданной таблично

4 Численное интегрирование

Если функция $f(x)$ непрерывна на отрезке $x \in [a, b]$ и известна ее первообразная $F(x)$, то определенный интеграл от этой функции в пределах от a до b может быть вычислен по формуле Ньютона – Лейбница:

$$\int_a^b f(x)dx = F(b) - F(a),$$

где $F'(x) = f(x)$ – первообразная подынтегральной функции $f(x)$.

Численное значение интеграла – это площадь криволинейной трапеции, ограниченной линиями графика функции и осью абсцисс Ox (выделенная область на рисунке 4).

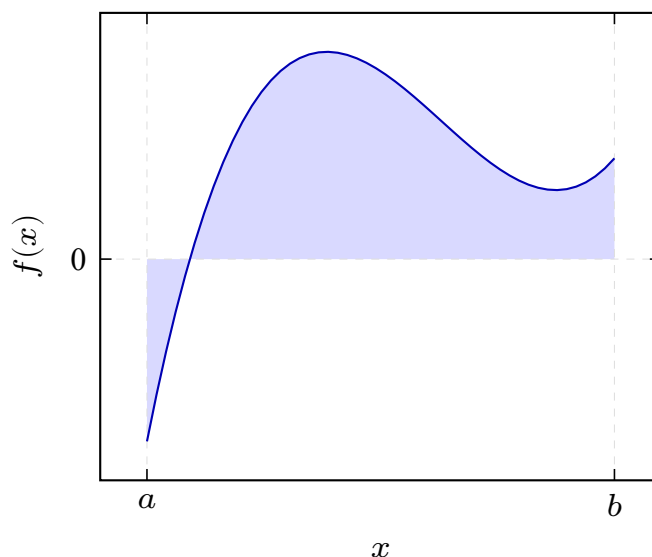


Рисунок 4 – Геометрический смысл определенного интеграла

Однако во многих случаях первообразная функция $F(x)$ не может быть найдена с помощью элементарных средств или является слишком сложной, поэтому вычисление определенного интеграла может быть затруднительным или даже практически невозможным.

Кроме того, на практике подынтегральная функция $f(x)$ часто задается таблично и тогда само понятие первообразной теряет смысл. Аналогичные вопросы возникают при вычислении кратных интегралов. Поэтому важное значение имеют приближенные и в первую очередь численные методы вычисления определенных интегралов.

Задача численного интегрирования функции заключается в вычислении значения определенного интеграла на основании ряда значений подынтегральной функции $f(x)$.

Обычный прием численного вычисления интеграла состоит в том, что данную функцию $f(x)$ на рассматриваемом отрезке $x \in [a, b]$ заменяют интерполирующей или аппроксимирующей функцией $\varphi(x)$ простого вида (например, полиномом), а затем приближенно полагают:

$$\int_a^b f(x)dx \approx \int_a^b \varphi(x)dx$$

Далее рассматриваются способы приближенного вычисления определенных интегралов вида:

$$I = \int_a^b \varphi(x)dx,$$

основанные на замене интеграла конечной суммой:

$$I \approx \sum_{i=0}^n c_i \cdot \varphi(x_i),$$

где c_i – числовые коэффициенты квадратурной формулы; x_i – узлы квадратурной формулы, т.е. точки отрезка $[a, b]$, $(i = 0, 1, \dots, n)$.

На основании свойств определенных интегралов, I можно представить в виде суммы интегралов по частичным отрезкам:

$$\int_a^b f(x)dx = \sum_{i=1}^n \int_{x_{i-1}}^{x_i} f(x)dx$$

Поэтому, для построения формулы численного интегрирования на всем отрезке $[a, b]$ достаточно построить квадратурную формулу на частичном отрезке $[x_{i-1}, x_i]$ для интеграла:

$$S_i = \int_{x_{i-1}}^{x_i} f(x)dx$$

4.1 Формула прямоугольников

В методе прямоугольников на частичном отрезке подынтегральная функция заменяется полиномом нулевой степени, то есть константу:

$$f(x) \approx L_0(x) = \text{const}$$

С геометрической точки зрения, в методе прямоугольников площадь криволинейной трапеции (интеграл от функции) на частичном отрезке заменяется площадью прямоугольника, ширина которого будет определяться расстоянием между соответствующими соседними узлами интегрирования, а высота – значением подынтегральной функции в этих узлах.

В зависимости от выбора узла сетки $\{x_i\}$ для аппроксимации подынтегральной функции $f(x)$ на частичном отрезке $[x_{i-1}, x_i]$ различают левую и правую формулы прямоугольников: если в качестве значения аппроксимирующего полинома выбирается значение подынтегральной функции на левом конце отрезка $L_0 \approx f(x_{i-1}) = y_{i-1}$ (рисунок 5), то справедлива левая формула прямоугольников:

$$S_i^- \approx \int_{x_{i-1}}^{x_i} L_0(x) dx = y_{i-1} \cdot (x_i - x_{i-1}),$$

а если значение аппроксимирующего полинома соответствует значению подынтегральной функции на правом конце частичного отрезка $L_0 \approx f(x_i) = y_i$ (рисунок 5), то справедлива правая формула прямоугольников :

$$S_i^+ \approx \int_{x_{i-1}}^{x_i} L_0(x) dx = y_i \cdot (x_i - x_{i-1}),$$

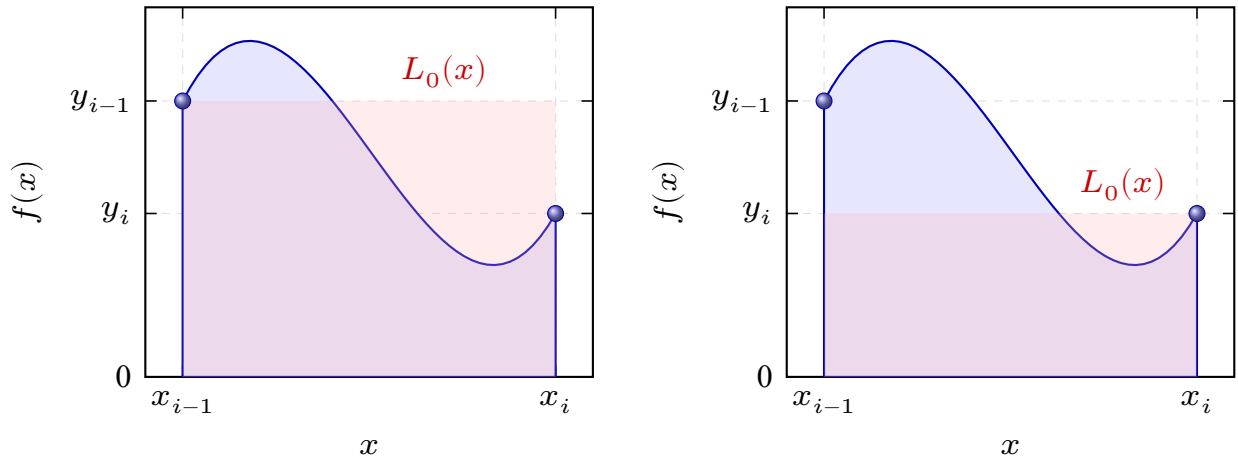


Рисунок 5 – График подынтегральной функции $f(x)$ и аппроксимирующего полинома $L_0(x)$ на частичном отрезке для формулы прямоугольников

4.2 Формула трапеций

Квадратурная *формула трапеций* является следствием замены на частичном отрезке подынтегральной функции интерполяционным полиномом первой степени $f(x) \approx L_1(x)$, построенным по множеству узлов сетки $\{x_{i-1}, x_i\}$:

$$L_1(x) = \frac{x - x_i}{x_{i-1} - x_i} \cdot y_{i-1} + \frac{x - x_{i-1}}{x_i - x_{i-1}} \cdot y_i.$$

Интегрирование интерполяционного полинома Лагранжа на частичном отрезке определяет формулу трапеций:

$$S_i \approx \int_{x_{i-1}}^{x_i} L_1(x) dx = \frac{y_i + y_{i-1}}{2} \cdot (x_i - x_{i-1})$$

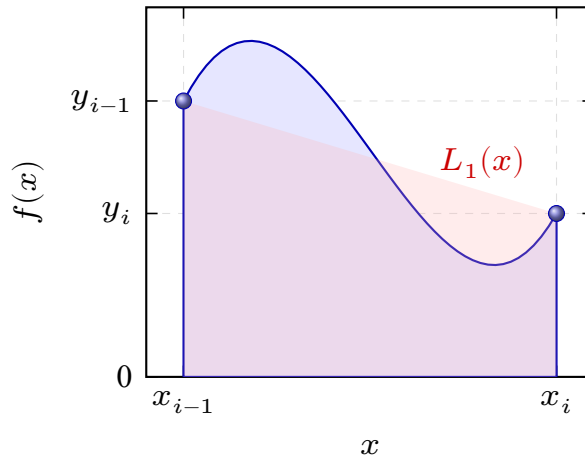


Рисунок 6 – График подынтегральной функции $f(x)$ и аппроксимирующего полинома $L_1(x)$ на частичном отрезке для формулы трапеций

4.3 Формула Симпсона

На частичном отрезке $[x_{i-1}, x_i]$ квадратурная *формула Симпсона* является следствием аппроксимации подынтегральной функции $f(x)$ интерполяционным полиномом Лагранжа второй степени $f(x) \approx L_2(x)$, который построен по узлам сетки $\{x_{i-1}, x_{i-1/2}, x_i\}$:

$$\begin{aligned}
 L_2(x) = & \frac{(x - x_{i-1/2}) \cdot (x - x_i)}{(x_{i-1/2} - x_{i-1}) \cdot (x_i - x_{i-1})} \cdot y_{i-1} + \\
 & + \frac{(x - x_{i-1}) \cdot (x_i - x)}{(x_{i-1/2} - x_{i-1}) \cdot (x_i - x_{i-1/2})} \cdot y_{i-1/2} + \\
 & + \frac{(x - x_{i-1}) \cdot (x - x_{i-1/2})}{(x_i - x_{i-1}) \cdot (x_i - x_{i-1/2})} \cdot y_i
 \end{aligned}$$

где $x_{i-1/2}$ – узел вспомогательной сетки, расположенный между узлами основной сетки $x_{i-1} < x_{i-1/2} < x_i$

Выражение для полинома Лагранжа в каноническом виде:

$$L_2(x) = c_0 + c_1 \cdot x + c_2 \cdot x^2,$$

где c_0, c_1, c_2 – коэффициенты при соответствующих степенях x интерполяционного полинома Лагранжа $L_2(x)$ в пределах частичного отрезка $[x_{i-1}, x_{i+1}]$.

Интегрирование интерполяционного полинома Лагранжа $L_2(x)$ на частич-

ном отрезке $x \in [x_{i-1}, x_{i+1}]$ определяет формулу Симпсона:

$$S_i \approx \int_{x_{i-1}}^{x_{i+1}} L_2(x) dx = c_0 \cdot (x_{i+1} - x_{i-1}) + c_1 \cdot \frac{x_{i+1}^2 - x_{i-1}^2}{2} + c_2 \cdot \frac{x_{i+1}^3 - x_{i-1}^3}{3}.$$

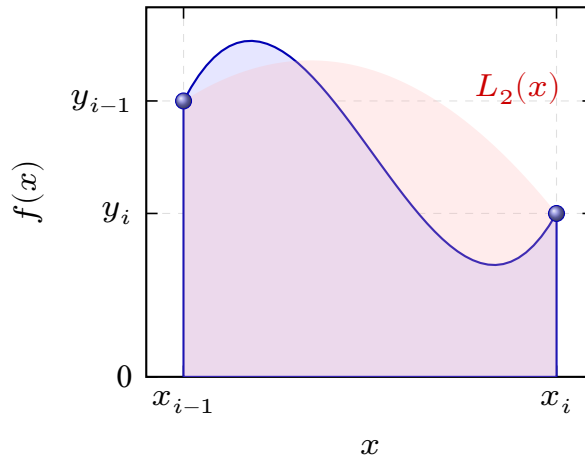


Рисунок 7 – График подынтегральной функции $f(x)$ и аппроксимирующего полинома $L_2(x)$ на частичном отрезке для формулы Симпсона

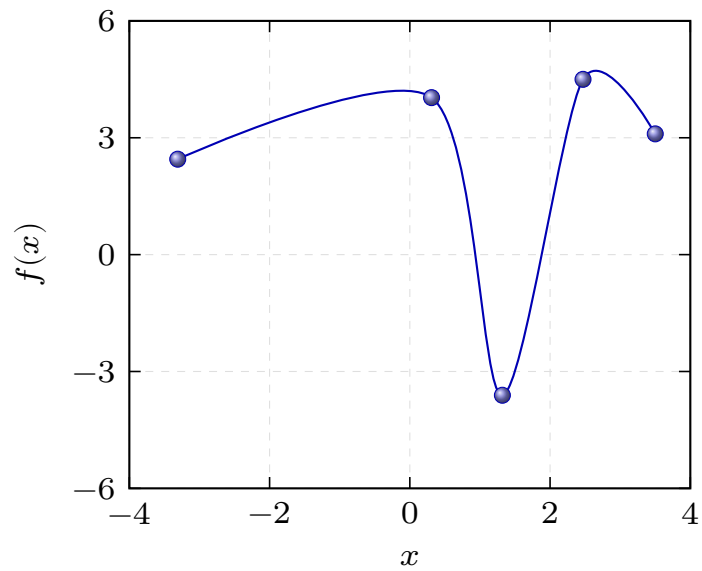
4.4 Численное интегрирования функции заданной таблично

На множестве узлов сетки $\{x_i\}$ определены значения некоторой функции $\{y_i\} = f(x_i)$:

Таблица 7 – Таблично заданная функциональная зависимость

| i | 0 | 1 | 2 | 3 | 4 |
|-------|-------|------|-------|------|------|
| x_i | −3.31 | 0.31 | 1.32 | 2.47 | 3.50 |
| y_i | 2.45 | 4.03 | −3.61 | 4.50 | 3.10 |

1) Построим график функции $f(x)$ заданной таблично.



- 2) Воспользуемся левой и правой формулами прямоугольников для нахождения численного значения интеграла от функции $f(x)$, заданной таблично на отрезке $x \in [x_0, x_4]$. Для этого разобьем весь отрезок интегрирования на частичные отрезки:

$$[x_0, x_4] = [x_0, x_1] \cup [x_1, x_2] \cup [x_2, x_3] \cup [x_3, x_4]$$

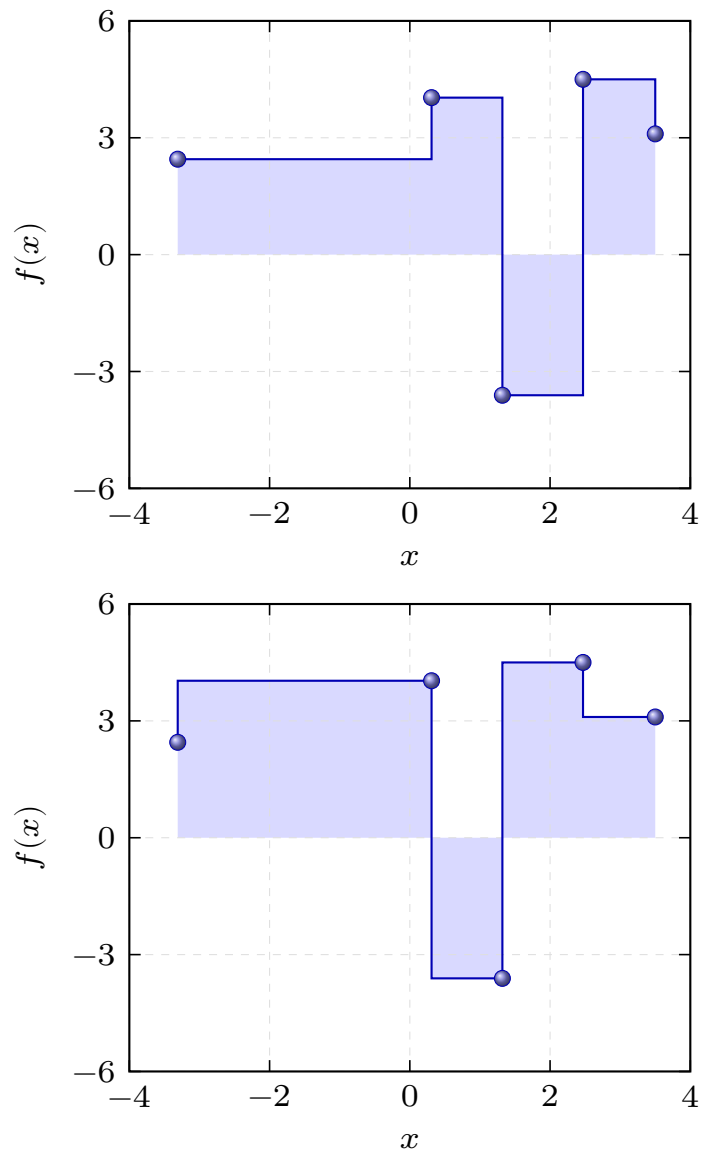


Рисунок 8 – Использование квадратурных формул левых и правых прямоугольников

На каждом частичном отрезке квадратурная формула является следствием замены подынтегральной функции $f(x)$ интерполяционным полиномом нулевой степени $f(x) \approx L_0(x) = \text{const}$, построенным по узлам $\{x_{i-1}, x_i\}$.

По методу прямоугольников, определим значение интеграла на каждом ча-

стичном отрезке (левые прямоугольники):

$$\begin{aligned} S_1^- &= y_0 \cdot (x_1 - x_0) = 2.45 \cdot (0.31 - (-3.31)) \approx 8.87 \\ S_2^- &= y_1 \cdot (x_2 - x_1) = 4.03 \cdot (1.32 - 0.31) \approx 4.07 \\ S_3^- &= y_2 \cdot (x_3 - x_2) = -3.61 \cdot (2.47 - 1.32) \approx -4.15 \\ S_4^- &= y_3 \cdot (x_4 - x_3) = 4.5 \cdot (3.50 - 2.47) \approx 4.64 \end{aligned}$$

(правые прямоугольники):

$$\begin{aligned} S_1^+ &= y_1 \cdot (x_1 - x_0) = 4.03 \cdot (0.31 - (-3.31)) \approx 14.59 \\ S_2^+ &= y_2 \cdot (x_2 - x_1) = -3.61 \cdot (1.32 - 0.31) \approx -3.65 \\ S_3^+ &= y_3 \cdot (x_3 - x_2) = 4.50 \cdot (2.47 - 1.32) \approx 5.18 \\ S_4^+ &= y_4 \cdot (x_4 - x_3) = 3.10 \cdot (3.50 - 2.47) \approx 3.19 \end{aligned}$$

Значение интегралов I^- и I^+ на всем отрезке интегрирования $[x_0, x_4]$:

$$\begin{aligned} I^- &= S_1^- + S_2^- + S_3^- + S_4^- = 8.87 + 4.07 - 4.15 + 4.64 = 13.43 \\ I^+ &= S_1^+ + S_2^+ + S_3^+ + S_4^+ = 14.59 - 3.65 + 5.18 + 3.19 = 19.31 \end{aligned}$$

- 3) Рассмотрим *метод трапеций* для нахождения численного значения интеграла от функции $f(x)$, заданной таблично на отрезке $x \in [x_0, x_4]$. Разобьем весь отрезок интегрирования на частичные отрезки:

$$[x_0, x_4] = [x_0, x_1] \cup [x_1, x_2] \cup [x_2, x_3] \cup [x_3, x_4]$$

На каждом частичном отрезке квадратурная формула является следствием замены подынтегральной функции $f(x)$ интерполяционным полиномом Лагранжа первой степени $f(x) \approx L_1(x)$, построенным по узлам $\{x_{i-1}, x_i\}$, т.е. прямой соединяющей два соседних узла.

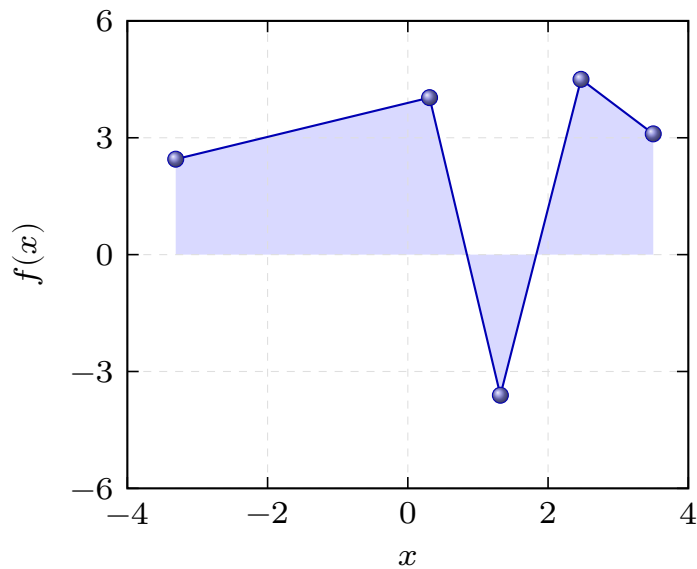


Рисунок 9 – Использование квадратурных формул трапеций

По методу трапеций, определим значение интеграла на каждом частичном отрезке:

$$S_1 = \frac{y_1 + y_0}{2} \cdot (x_1 - x_0) = \frac{4.03 + 2.45}{2} \cdot (0.31 - (-3.31)) \approx 11.73$$

$$S_2 = \frac{y_2 + y_1}{2} \cdot (x_2 - x_1) = \frac{-3.61 + 4.03}{2} \cdot (1.32 - 0.31) \approx 0.21$$

$$S_3 = \frac{y_3 + y_2}{2} \cdot (x_3 - x_2) = \frac{4.50 - 3.61}{2} \cdot (2.47 - 1.32) \approx 0.51$$

$$S_4 = \frac{y_4 + y_3}{2} \cdot (x_4 - x_3) = \frac{3.10 + 4.50}{2} \cdot (3.50 - 2.47) \approx 3.91$$

Определим интеграл I на всем отрезке интегрирования $[x_0, x_4]$, воспользовавшись свойством аддитивности интеграла:

$$I = S_1 + S_2 + S_3 + S_4 = 11.73 + 0.21 + 0.51 + 3.91 = 16.37.$$

- 4) Рассмотрим *метод Симпсона* для нахождения численного значения интеграла от функции $f(x)$, заданной таблично на отрезке $x \in [x_0, x_4]$.

Разделим всё множество узлов сетки $\{x_i\}$, в которых известны значения функции $\{y_i\}$, на основные и вспомогательные узлы:

Таблица 8 – Таблично заданная функциональная зависимость

| i | 0 | $1 - 1/2$ | 1 | $1 + 1/2$ | 2 |
|-------|-------|-----------|-------|-----------|------|
| x_i | -3.31 | 0.31 | 1.32 | 2.47 | 3.50 |
| y_i | 2.45 | 4.03 | -3.61 | 4.50 | 3.10 |

Разобьем весь отрезок интегрирования на частичные отрезки:

$$[x_0, x_2] = [x_0, x_1] \cup [x_1, x_2].$$

В пределах первого частичного отрезка $[x_0, x_1]$ построим интерполяционный полином Лагранжа $L_2(x)$ по узлам сетки $x_0 = -3.31$, $x_{1-1/2} = 0.31$, $x_1 = 1.32$:

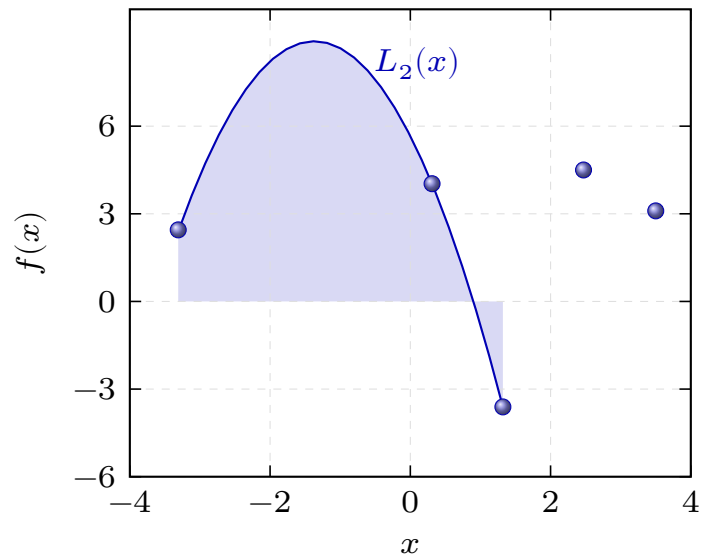
$$\begin{aligned}
 L_2(x) = & \frac{(x - 0.31)(x - 1.32)}{((-3.31 - 0.31)(-3.31 - 1.32))} \cdot 2.45 + \\
 & + \frac{(x - (-3.31))(x - 1.32)}{(0.31 - (-3.31))(0.31 - 1.32)} \cdot 4.03 + \\
 & + \frac{(x - (-3.31))(x - 0.31)}{(1.32 - (-3.31))(1.32 - 0.31)} \cdot (-3.61)
 \end{aligned}$$

После алгебраических преобразований запишем интерполяционный полином в каноническом виде:

$$L_2(x) = 5.66 - 4.74 \cdot x - 1.73 \cdot x^2$$

Определим интеграл от интерполяционного полинома $L_2(x)$ на первом частичном отрезке:

$$I_1 = \int_{x_0}^{x_1} L_2(x) dx = \int_{-3.31}^{1.32} (5.66 - 4.74 \cdot x - 1.73 \cdot x^2) dx = 25.88$$



В пределах второго частичного отрезка $[x_1, x_2]$ построим интерполяционный полином Лагранжа $L_2(x)$ по узлам сетки $x_1 = 1.32$, $x_{1+1/2} = 2.47$, $x_2 = 3.50$:

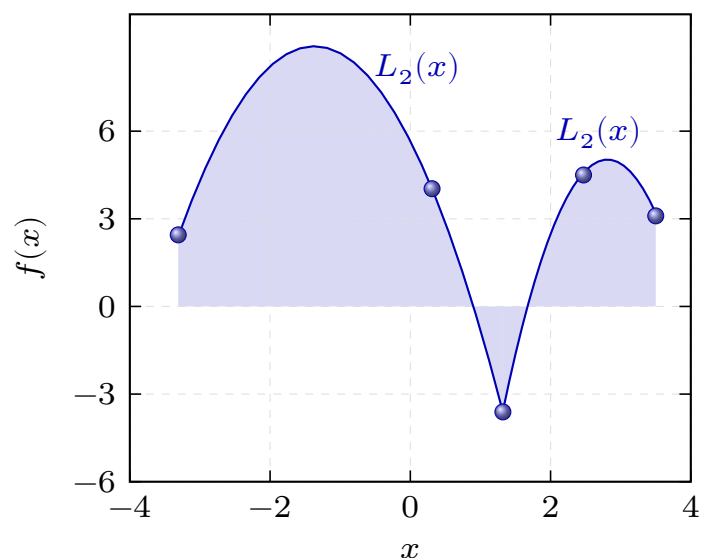
$$\begin{aligned} L_2(x) &= \frac{(x - 2.47)(x - 3.50)}{(1.32 - 2.47)(1.32 - 3.50)} \cdot (-3.61) + \\ &+ \frac{(x - 1.32)(x - 3.50)}{(2.47 - 1.32)(2.47 - 3.50)} \cdot 4.50 + \\ &+ \frac{(x - 1.32)(x - 2.47)}{(3.50 - 1.32)(3.50 - 2.47)} \cdot 3.10 \end{aligned}$$

После тривиальных алгебраических преобразований:

$$L_2(x) = -25.56 + 21.76 \cdot x - 3.87 \cdot x^2$$

Определим интеграл от интерполяционного полинома $L_2(x)$ на втором частичном отрезке:

$$I_2 = \int_{x_1}^{x_2} L_2(x) dx = \int_{1.32}^{3.50} (-25.56 + 21.76 \cdot x - 3.87 \cdot x^2) dx = 6.13$$



Определим интеграл всем отрезке $[x_0, x_2]$ воспользовавшись свойством аддитивности интеграла:

$$I = I_1 + I_2 = 25.88 + 6.13 = 32.01$$

- 5) Сравнивая численные значения определенного интеграла рассчитанные по методам прямоугольников, трапеций и Симпсона, можно сделать вывод о том, что рассчитанные значения различаются.

Таблица 9 – Численные значения интегралов

| Метод интегрирования | Значение интеграла |
|------------------------|--------------------|
| Левых прямоугольников | 13.43 |
| Правых прямоугольников | 19.31 |
| Трапеций | 16.37 |
| Симпсона | 32.01 |

Значение определенного интеграла от функции заданной таблично, рассчитанное по методу Симпсона является наибольшим, а значение рассчитанное по методу левых прямоугольников – наименьшее.

5 Решение нелинейных уравнений

Пусть задана функция $f(x)$ действительного переменного и необходимо найти корни уравнения или, что то же самое, нули функции $f(x)$:

$$f(x) = 0. \quad (11)$$

На примере алгебраического многочлена известно, что нули $f(x)$ могут быть как действительными, так и комплексными числами. Поэтому *более точная* постановка задачи состоит в нахождении корней уравнения, расположенных в заданной области комплексной плоскости. Можно рассматривать также задачу о нахождении действительных корней уравнения, которые расположены в пределах заданного отрезка $x \in [a, b]$.

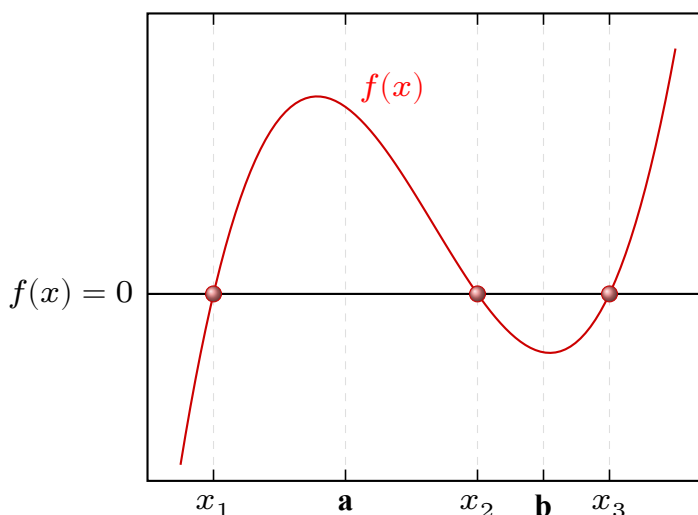


Рисунок 10 – График функции $y = f(x)$

На рисунке (10) представлены x_1 , x_2 и x_3 – действительные корни уравнения (11), т.е. $f(x_1) = 0$, $f(x_2) = 0$, $f(x_3) = 0$

Задача нахождения корней уравнения $f(x) = 0$ обычно решается в два этапа:

- 1) На первом этапе изучается расположение корней (в общем случае на комплексной плоскости) и проводится их разделение, т. е. *выделяются области* в комплексной плоскости, *содержащие только один корень*. Кроме того, изучается вопрос о кратности корней. Тем самым находятся некоторые начальные приближения для корней уравнения.
- 2) На втором этапе, *используя заданное начальное приближение*, строится итерационный процесс.

рационный процесс, позволяющий *уточнить значение отыскиваемого корня*.

Следует отметить, что не существует каких-то общих регулярных приемов решения задачи о расположении корней произвольной функции $f(x)$.

Численные методы решения нелинейных уравнений являются, как правило, итерационными методами, которые предполагают задание достаточно близких к искомому решению начальных данных.

5.1 Итервальный метод

Итервальный метод поиска корня уравнения $f(x) = 0$ состоит следующим:

- 1) Область поиска корня $[a, b]$ разбивается на заранее заданное количество интервалов N :

$$[a, b] = [a, x_1] \cup [x_1, x_2] \cup \dots \cup [x_i, x_{i+1}] \cup \dots [x_N, b]$$

- 2) Вычисляется таблица значений функции $\{f(x_i)\}$ на границах этих интервалов $\{x_i\}$.
- 3) Проводится последовательный перебор таблицы значений функции $\{f(x_i)\}$ ($i = 1, 2, \dots, N$).
- 4) Если при некотором i значения функции $f(x_i)$ и $f(x_{i+1})$ имеют разные знаки, то это означает, что на интервале $x \in (x_i, x_{i+1})$ имеет по крайней мере один действительный корень уравнения $f(x) = 0$.
- 5) В качестве новой, более узкой ($|x_i, x_{i+1}| < |a, b|$), области поиска выбирается отрезок (x_i, x_{i+1}) , т.е. полагают

$$x_i = a, \quad x_{i+1} = b$$

и с помощью аналогичной процедуры (1) процесс поиска корня уравнения $f(x) = 0$ повторяют до тех пор пока, область поиска не станет меньше заранее заданной величины ϵ (погрешности поиска корня уравнения):

$$(b - a) < \epsilon$$

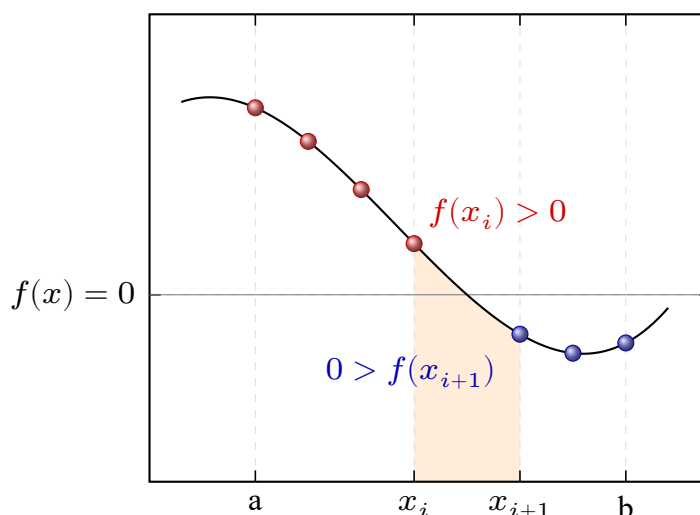


Рисунок 11 – Иллюстрация итервального метода поиска корня уравнения

5.2 Метод бисекции

Метод бисекции основан на теореме *Больцано-Коши* (теорема о промежуточном значении): если непрерывная функция $f(x)$, определённая на вещественном интервале $[a, b]$, принимает два различных значения $f(a) \neq f(b)$, тогда существует такое $c \in [a, b]$, что эта функция в этой точке принимает промежуточное значение $f(a) \leq f(c) \leq f(b)$.

Следствие теоремы Больцано-Коши (теорема о нуле непрерывной функции): если функция $f(x)$ непрерывна на некотором отрезке $[a, b]$ и на концах этого отрезка принимает значения $f(a)$ и $f(b)$ противоположных знаков, то существует точка x_0 , в которой значение функции равно нулю $f(x_0) = 0$.

Если непрерывная функция *строго монотонна* на отрезке $[a, b]$, т.е. для любого $\forall x \in [a, b]$ выполняется условие $f'(x) > 0$ либо $f'(x) < 0$, то в соответствие со следствием теоремы Больцано-Коши в пределах отрезка $[a, b]$ существует *единственный* корень уравнения $f(x) = 0$.

Метод бисекции (деления пополам) является регулярным способом поиска действительного корня уравнения $f(x) = 0$, однако для реализации этого метода необходимо *правильно выбрать область поиска*, т.е. начальный отрезок $[a, b]$ на концах которого функция $f(x)$ принимает значения разных знаков $f(a) \cdot f(b) < 0$ и в пределах этого отрезка строго монотонна $f'(x) < 0$ либо $f'(x) > 0$.

Алгоритм метода деления отрезка пополам (метод бисекции):

1) Область поиска корня уравнения отрезок $[a, b]$ делится пополам:

$$c = \frac{a + b}{2}$$

2) Вычисляется значение функции в середине отрезка $f(c)$.

3) Проводится сравнение знаков функции в середине отрезка $f(c)$ и, например, на левом конце отрезка $f(a)$:

1) если $f(a) \cdot f(c) < 0$, функция $f(x)$ на концах отрезка $[a, c]$ принимает значения разных знаков, следовательно, искомый корень уравнения $f(x) = 0$ находится внутри отрезка $[a, c]$, поэтому правый конец отрезка “переносится” в его середину.

2) если $f(a) \cdot f(c) > 0$, функция $f(x)$ на концах отрезка $[a, c]$ принимает значения одного знака, следовательно, искомый корень уравнения $f(x) = 0$ находится внутри отрезка $[c, b]$, поэтому левый конец отрезка “переносится” в его середину.

$$\text{sign}(f(a) \cdot f(c)) = \begin{cases} < 0, & b = c \\ > 0, & a = c \end{cases}$$

Таким образом, область поиска корня уравнения $f(x) = 0$ “сужается наполовину”.

4) Процесс вычислений (1)–(3) повторяется до тех пока, длина вновь полученного интервала $[a, b]$ станет меньше заранее заданного числа ϵ (погрешности поиска корня уравнения):

$$(b - a) < \epsilon$$

В качестве корня уравнения x_0 приближенно принимаются середину последнего полученного интервала $[a, b]$.

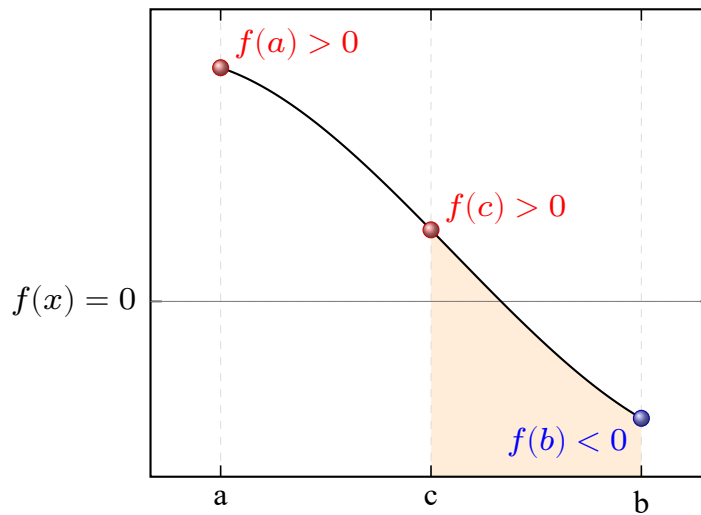


Рисунок 12 – Иллюстрация метода бисекции (деления отрезка пополам):

$f(a) \cdot f(c) > 0$, поэтому новая область поиска корня отрезок $[c, b]$

Следует отметить, что если условие строгой монотонности для функции $f(x)$ на отрезке $[a, b]$ не выполняется и на этом отрезке имеется несколько корней, то указанный итерационный процесс сойдется к одному из корней, но *заранее неизвестно, к какому именно.*

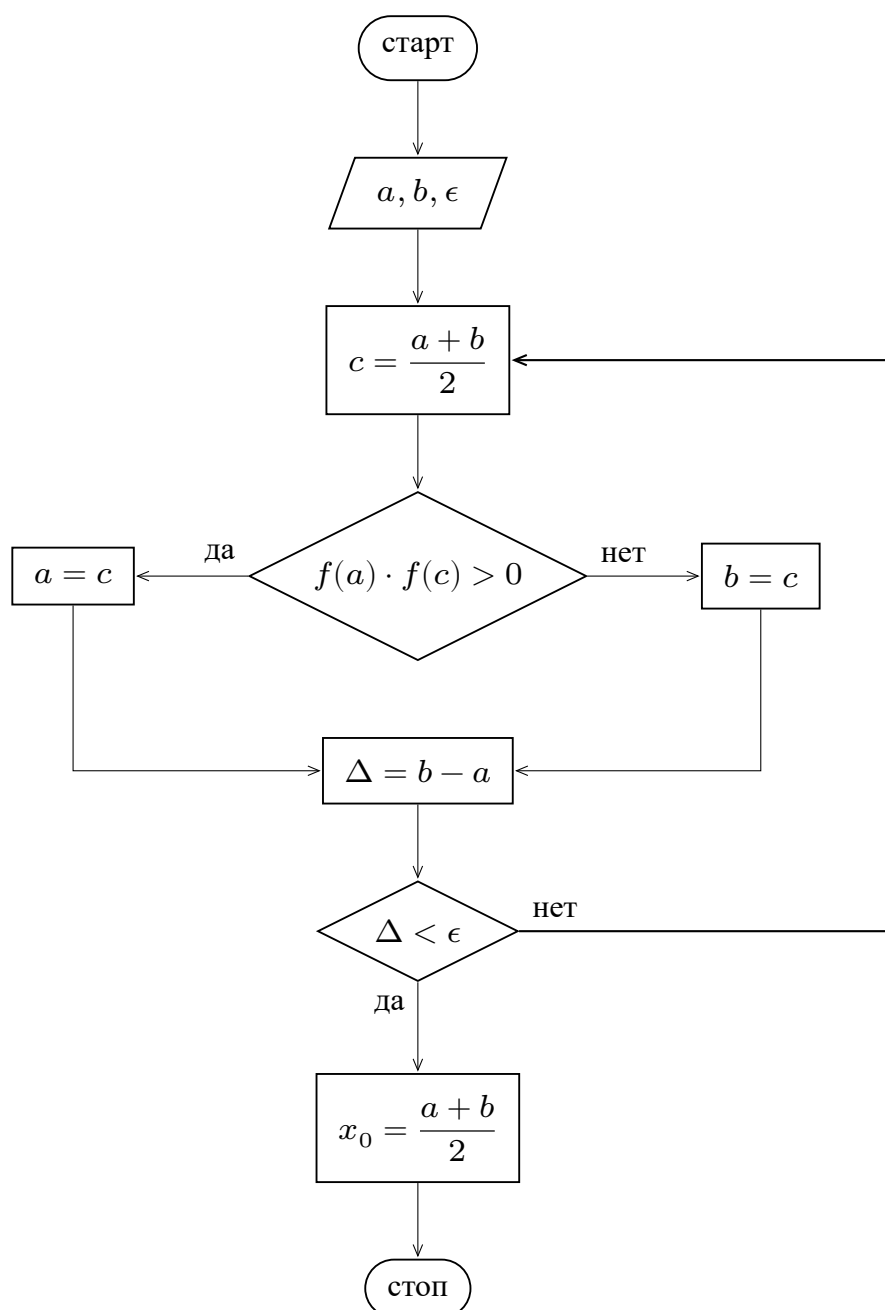


Рисунок 13 – Блок-схема алгоритма поиска решения x_0 нелинейного уравнения $f(x) = 0$

5.3 Метод выделения корней

Один из недостатков интервального метода и метода бисекции является сходимость итерационного процесса к заранее неизвестному корню уравнения $f(x) = 0$. Этот недостаток можно устранить удалением уже найденного корня.

Если x_1 простой корень уравнения $f(x) = 0$ и функция $f(x)$ непрерывна

по Липшицу, то вспомогательная функция

$$g(x) = \frac{f(x)}{(x - x_1)}$$

непрерывна, причем все нули функций $f(x)$ и $g(x)$ совпадают, за исключением x_1 , так как $g(x_1) \neq 0$.

Поэтому найденный корень x_1 можно удалить, т.е. перейти в процессе поиска корня уравнения $f(x) = 0$ от функции $f(x)$ к функции $g(x)$. Тогда процесс нахождения остальных корней уравнения сведется к нахождению корней $g(x) = 0$.

Когда найден какой-нибудь новый корень x_2 уравнения $g(x) = 0$, то этот корень тоже можно удалить, вводя новую вспомогательную функцию:

$$\varphi(x) = \frac{g(x)}{(x - x_2)} = \frac{f(x)}{(x - x_1)(x - x_2)}$$

Таким образом, можно последовательно найти все корни исходного уравнения $f(x) = 0$.

В любом методе поиска корней уравнения $f(x)$ окончательные итерации вблизи определяемого корня рекомендуется делать не по функциям типа $g(x)$, а по исходной функции $f(x)$. Последние итерации, вычисленные по функции $g(x)$, используются при этом в качестве нулевого приближения.

5.4 Численное решение нелинейного уравнения методом бисекции

На отрезке $x \in [-3, 5]$ задана непрерывная функция:

$$f(x) = \tanh(x) \cdot (1 + \cos(x)) - \frac{1}{2}$$

С помощью метода бисекции найдем первый положительный корень $x_1 > 0$ нелинейного уравнения $f(x) = 0$.

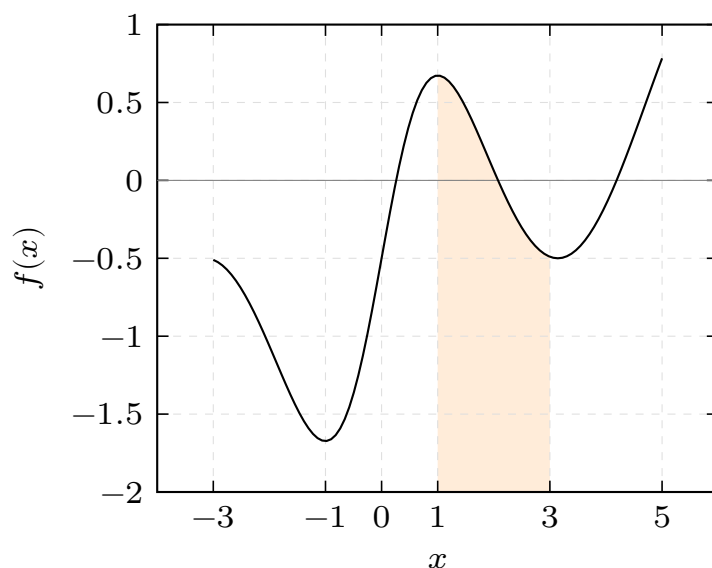


Рисунок 14 – График функции $f(x) = \tanh(x) \cdot (1 + \cos(x)) - \frac{1}{2}$

На основе анализа графика функции $f(x)$ (рисунок 14) выбираем область поиска первого положительного корня уравнения ($x > 0$), на границах которой функция $f(x)$ принимает значения разных знаков $f(a) \cdot f(b) < 0$ и в пределах этого области строго монотонна $f'(x) < 0$.

Таким требованиям удовлетворяет отрезок $x \in [1, 3]$ (выделенная область на графике), так как функция на отрезке монотонна $f'(x) < 0$ и на концах отрезка принимает значения разных знаков:

$$f(1) = \tanh(1) \cdot (1 + \cos(1)) - \frac{1}{2} \approx 0.67 > 0$$

$$f(3) = \tanh(3) \cdot (1 + \cos(3)) - \frac{1}{2} \approx -0.49 < 0$$

Зададим погрешность поиска корня уравнения $\varepsilon = 0.1$ и используем метод бисекции для поиска первого положительного корня уравнения:

- 1) Область поиска корня уравнения отрезок $[1, 3]$ делится пополам и вычисляется значение функции в середине отрезка $f(c)$:

$$c = \frac{1 + 3}{2} = 2$$

$$f(c) = f(2) = \tanh(2) \cdot (1 + \cos(2)) - \frac{1}{2} \approx 0.06285 > 0$$

Проводится сравнение знаков функции в середине отрезка $f(c)$ и на левом

конце отрезка $f(a)$:

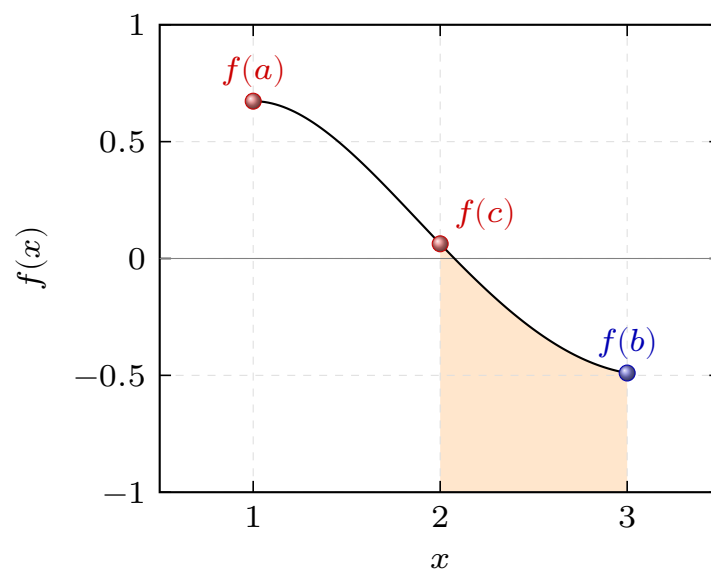
$$f(a) \cdot f(c) = f(1) \cdot f(2) = 0.67 \cdot 0.06285 = 0.0423 > 0$$

Следовательно, искомый корень уравнения находится внутри отрезка $[2, 3]$.

Проведем сравнение длины отрезка и погрешности поиска корня:

$$|a, b| = b - a = 3 - 2 = 1 > \varepsilon = 0.1$$

Так как длина отрезка больше погрешности поиска, то итерационный процесс продолжаем.



- 2) Область поиска корня уравнения отрезок $[2, 3]$ делится пополам и вычисляется значение функции в середине отрезка $f(c)$:

$$c = \frac{2 + 3}{2} = 2.5$$

$$f(c) = f(2.5) = \tanh(2.5) \cdot (1 + \cos(2.5)) - \frac{1}{2} \approx -0.3038 < 0$$

Проводится сравнение знаков функции в середине отрезка $f(c)$ и на левом конце отрезка $f(a)$:

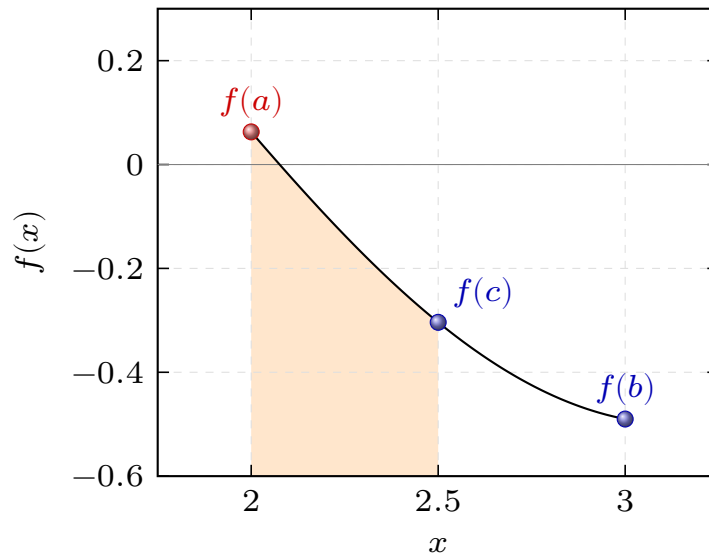
$$f(a) \cdot f(c) = f(2) \cdot f(2.5) = 0.6285 \cdot (-0.3038) = -0.019094 < 0$$

Следовательно, искомый корень уравнения находится внутри отрезка $[2, 2.5]$.

Проведем сравнение длины отрезка и погрешности поиска корня:

$$|a, b| = b - a = 2.5 - 2 = 0.5 > \varepsilon = 0.1$$

Так как длина отрезка больше погрешности поиска, то итерационный процесс продолжаем.



- 3) Область поиска корня уравнения отрезок $[2, 2.5]$ делится пополам и вычисляется значение функции в середине отрезка $f(c)$:

$$c = \frac{2 + 2.5}{2} = 2.25$$

$$f(c) = f(2.25) = \tanh(2.25) \cdot (1 + \cos(2.25)) - \frac{1}{2} \approx -0.13634 < 0$$

Проводится сравнение знаков функции в середине отрезка $f(c)$ и на левом конце отрезка $f(a)$:

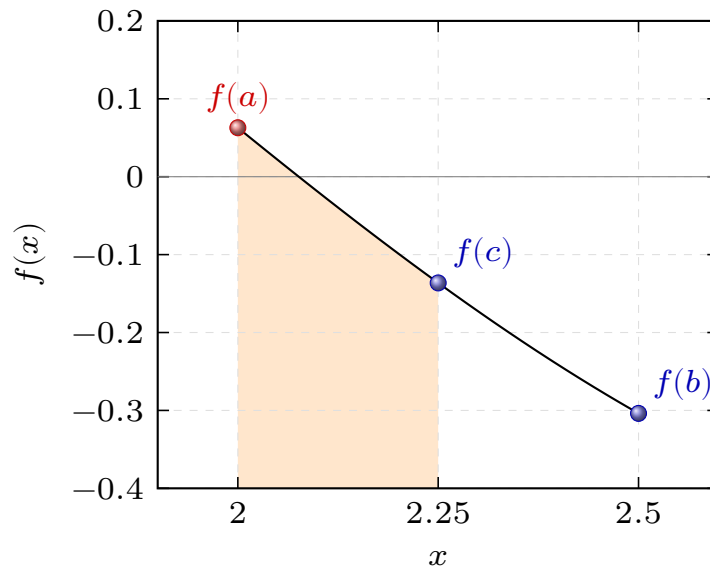
$$f(a) \cdot f(c) = f(2) \cdot f(2.25) = 0.6285 \cdot (-0.13634) = -0.00857 < 0$$

Следовательно, искомый корень уравнения находится внутри отрезка $[2, 2.25]$.

Проведем сравнение длины отрезка и погрешности поиска корня:

$$|a, b| = b - a = 2.25 - 2 = 0.25 > \varepsilon = 0.1$$

Длина отрезка меньше погрешности поиска, поэтому итерационный процесс продолжаем.



- 4) Область поиска корня уравнения отрезок $[2, 2.25]$ делится пополам и вычисляется значение функции в середине отрезка $f(c)$:

$$c = \frac{2 + 2.25}{2} = 2.125$$

$$f(c) = f(2.125) = \tanh(2.125) \cdot (1 + \cos(2.125)) - \frac{1}{2} \approx -0.03959 < 0$$

Проводится сравнение знаков функции в середине отрезка $f(c)$ и на левом конце отрезка $f(a)$:

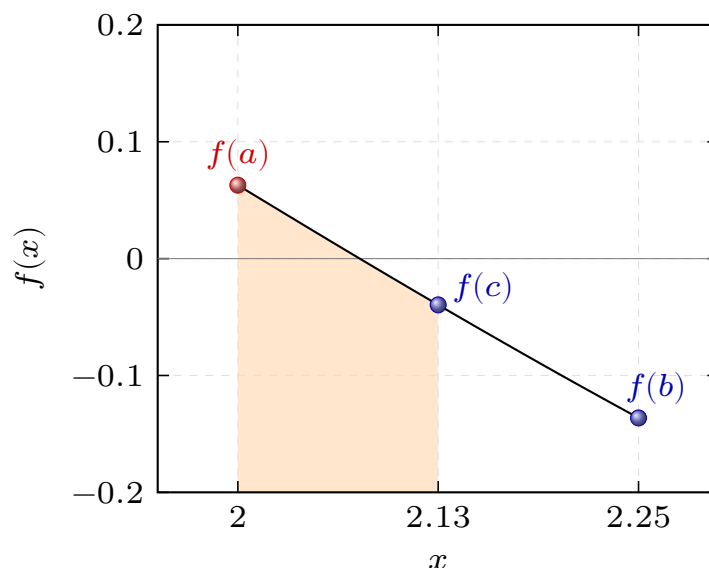
$$f(a) \cdot f(c) = f(2) \cdot f(2.125) = 0.6285 \cdot (-0.03959) = -0.002488 < 0$$

Следовательно, искомый корень уравнения находится внутри отрезка $[2, 2.125]$.

Проведем сравнение длины отрезка и погрешности поиска корня:

$$|a, b| = b - a = 2.125 - 2 = 0.125 > \varepsilon = 0.1$$

Длина отрезка меньше погрешности поиска, поэтому итерационный процесс продолжаем.



- 5) Область поиска корня уравнения отрезок $[2, 2.125]$ делится пополам и вычисляется значение функции в середине отрезка $f(c)$:

$$c = \frac{2 + 2.125}{2} = 2.0625$$

$$f(c) = f(2.0625) = \tanh(2.0625) \cdot (1 + \cos(2.0625)) - \frac{1}{2} \approx 0.01108 > 0$$

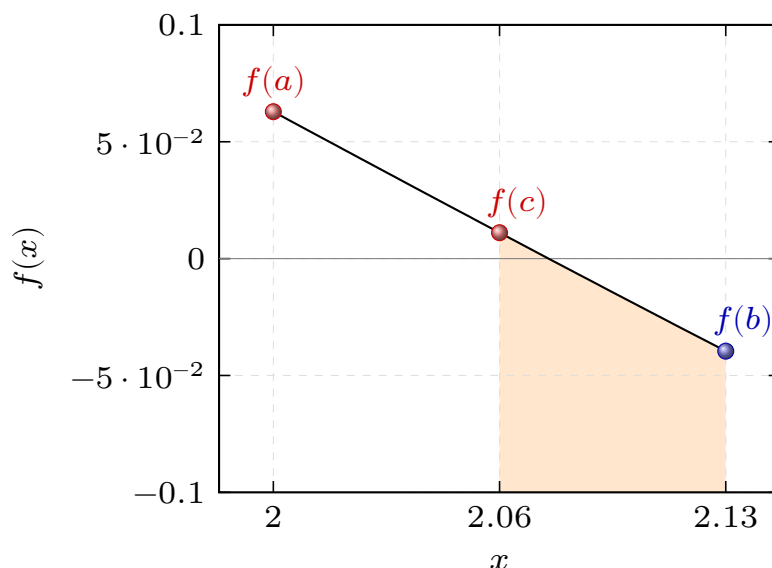
Проводится сравнение знаков функции в середине отрезка $f(c)$ и на левом конце отрезка $f(a)$:

$$f(a) \cdot f(c) = f(2) \cdot f(2.0625) = 0.6285 \cdot 0.01108 = 0.0007 > 0$$

Следовательно, искомый корень уравнения находится внутри отрезка $[2.0625, 2.125]$. Проведем сравнение длины отрезка и погрешности поиска корня:

$$|a, b| = b - a = 2.125 - 2.0625 = 0.0625 < \varepsilon = 0.1$$

Длина отрезка меньше погрешности поиска, поэтому итерационный процесс завершаем.



В качестве первого положительного корня x_1 уравнения $f(x) = 0$ приближенно выберем середину последнего полученного интервала $[a, b]$ и для контроля определим значение функции в точке $f(x_1)$ приближенного корня уравнения:

$$x_1 = \frac{a + b}{2} = \frac{2.0625 + 2.125}{2} = 2.09375$$

$$f(x_1) = f(2.09375) = \tanh(2.09375) \cdot (1 + \cos(2.09375)) - \frac{1}{2} \approx -0.01442$$

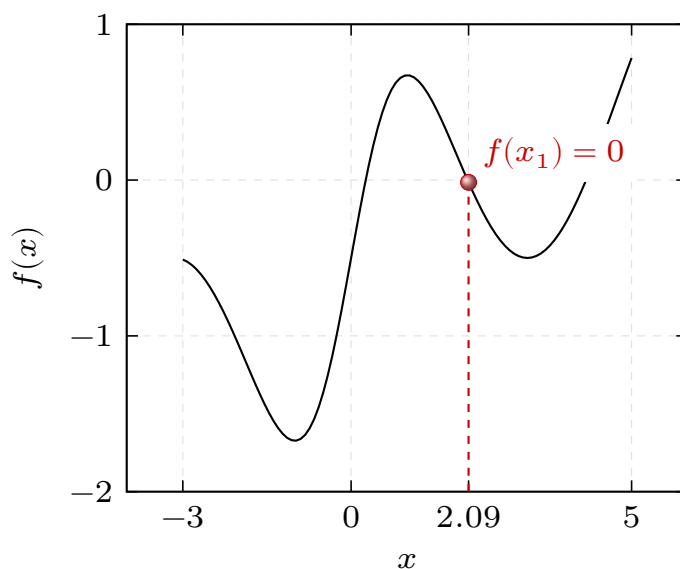


Рисунок 15 – График функции $f(x)$ (сплошная линия) и первый положительный корень $x_1 = 2.09375$ (маркер) уравнения $f(x) = 0$

6 Методы локальной оптимизации

Оптимизация – это задача нахождения экстремума (минимума или максимума) целевой функции в некоторой области конечномерного векторного пространства, ограниченной набором линейных и/или нелинейных равенств и/или неравенств.

Во многих практически важных случаях для целевой функции многих переменных $f(\mathbf{x})$ задача оптимизации может быть сформулирована в виде:

$$f(\mathbf{x}) \rightarrow \min,$$

где $\mathbf{x} = (x_1, x_2, \dots, x_n)$ – вектор неизвестных (управляющих параметров); \min – минимальное значение функции в ограниченной или неограниченной области изменения неизвестных.

Для нахождения абсолютного минимума целевой функции $f(\mathbf{x})$ существует только один способ: найти все локальные минимумы этой функции, сравнить их и выбрать из них тот, в котором функция принимает наименьшее значение.

6.1 Минимум функции одного переменного

Для функции одной переменной $f(x)$, задача нахождения минимума эквивалента задачи нахождения корней уравнения:

$$\frac{df(x)}{dx} = 0 \quad (12)$$

Эта одномерная задача нередко возникает в практических приложениях. Кроме того, большинство методов решения многомерных задач сводится к поиску одномерного минимума.

Предположим, что $f(x)$ задана и кусочно-непрерывна на отрезке $x \in [a, b]$, и имеет на этом отрезке (включая его концы) только один локальный минимум. Построим итерационный процесс, сходящийся к этому минимуму.

Вычислим значение функции на концах отрезка $x = a$ и $x = b$, а также в двух внутренних точках $x_1 < x_2$. Так как функция $f(x)$ имеет минимум на отрезке $x \in [a, b]$, то справедливо утверждение:

$$f(a) \geq f(x_1), \quad f(x_2) \leq f(b)$$

Сравним все четыре значения функции между собой $f(a)$, $f(x_1)$, $f(x_2)$ и $f(b)$ и выберем среди них наименьшее.

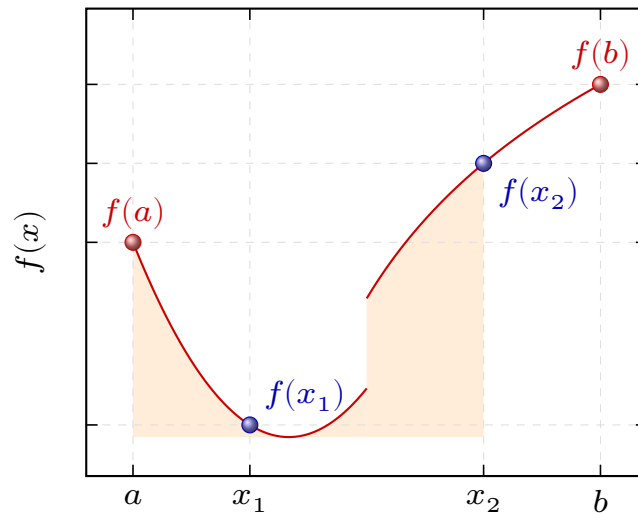


Рисунок 16 – График кусочно-непрерывной функции $y = f(x)$, имеющей минимум на отрезке $x \in [a, b]$

Из рисунка 16 видно, что наименьшее значение функция достигает в точке $x = x_1$:

$$f(x_1) < f(a) < f(x_2) < f(b)$$

Очевидно, что минимум функции $f(x)$ расположен в одном из прилегающих к точке $x = x_1$ отрезков, то есть минимум находится либо в пределах отрезка $[a, x_1]$, либо в $[x_1, x_2]$ (рисунок 16, выделенная область).

Поэтому на первом шаге итерационного процесса отбрасывается отрезок $[x_2, b]$, и для поиска минимума функции $f(x)$ рассматривается отрезок $[a, x_2]$, при этом область поиска минимума функции сужается:

$$|a - x_2| < |a - b|, \quad \text{так как} \quad x_2 < b.$$

Полагая $b = x_2$, на новом отрезке $[a, b]$ вновь необходимо выбрать две внутренние точки, вычислить в них и на концах отрезка значения функции $f(x)$, и сделать следующий шаг итерационного процесса.

Критерием остановки итерационного процесса является условие выполнения неравенства, которое гарантирует малость размера области поиска ми-

нимума по сравнению с заранее заданной погрешность метода:

$$(b - a) \leq \epsilon,$$

где ϵ – погрешность метода.

Симметричный метод поиска минимума функции одной переменной $f(x)$ основан на выборе внутренних точек x_1 и x_2 отрезка $[a, b]$, которые равноудалены от концов этого отрезка. Например, если точки x_1 и x_2 делят отрезок $[a, b]$ на три равные части (рисунок 17), то координаты этих точек могут быть определены из соотношений:

$$x_1 = a + \frac{b - a}{3} = \frac{2a + b}{3}, \quad x_2 = b - \frac{b - a}{3} = \frac{a + 2b}{3}.$$



Рисунок 17 – Схематическое изображение точек деления отрезка $[a, b]$

Оценка длины отрезка после первого итерационного шага составит:

$$\ell_1 = (b - a) - \frac{b - a}{3} = \frac{2}{3} \cdot (b - a),$$

после второго шага:

$$\ell_2 = \ell_1 - \frac{\ell_1}{3} = \frac{2}{3} \cdot \ell_1 = \left(\frac{2}{3}\right)^2 \cdot (b - a),$$

а после k -ого итерационного шага:

$$\ell_k = \left(\frac{2}{3}\right)^k \cdot (b - a).$$

Таким образом, чтобы погрешность вычисления ℓ_k была менее ϵ , для числа итераций k справедлива оценка:

$$\left(\frac{2}{3}\right)^k \cdot (b - a) \leq \epsilon \quad \rightarrow \quad k = \left\lceil \frac{\ln(b - a) - \ln(\epsilon)}{\ln(3) - \ln(2)} \right\rceil$$

Симметричный метод поиска минимума функции является аналогом ме-

тогда дихотомии для нахождения корня уравнения $f(x) = 0$. Метод применим к недифференцируемым функциям и всегда сходится. Следует отметить, что если на отрезке $[a, b]$ функция $f(x)$ имеет несколько локальных минимумов, то итерационный процесс сойдется к одному из этих минимумов, но не обязательно к наименьшему.

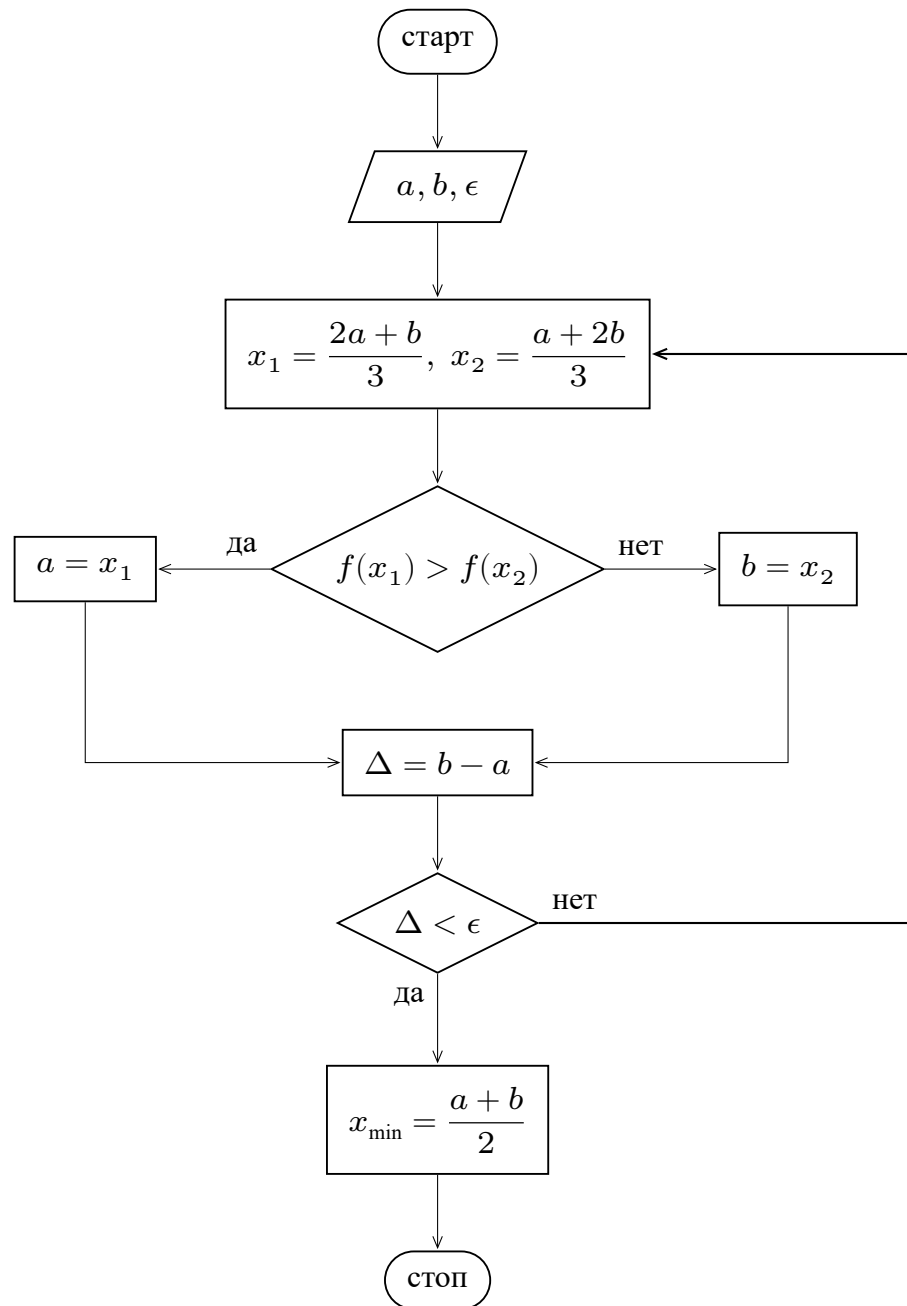


Рисунок 18 – Блок-схема алгоритма нахождения минимума функции одного переменного $x_{\min} = \arg \left(\min_{a \leq x \leq b} f(x) \right)$

6.2 Минимизация функций многих переменных

Задача безусловной минимизации (оптимизации) состоит в нахождении минимума или максимума функции $f(x)$ в отсутствие каких-либо ограничений на область изменения переменных задачи $x \in (-\infty, +\infty)$.

Большинство практических задач оптимизации содержит ограничения, которые обусловлены технико-экономическим смыслом решаемой задачи, однако многие алгоритмы решения задач с ограничениями предполагают сведение ее к последовательности задач безусловной оптимизации.

6.2.1 Спуск по координатам

Идея метода по координатного спуска заключается в том, что задача поиска минимума функции $f(x_1, x_2, \dots, x_n) \rightarrow \min$ разделяется на серию задач *одномерной минимизации* этой функции вдоль направления одной из координатных осей:

$$f(\textcolor{red}{x}_1, x_2 = \text{const}, \dots, x_n = \text{const}) \rightarrow \min$$
$$f(x_1 = \text{const}, \textcolor{red}{x}_2, \dots, x_n = \text{const}) \rightarrow \min$$
$$\dots\dots\dots$$
$$f(x_1 = \text{const}, x_2 = \text{const}, \dots, \textcolor{red}{x}_n) \rightarrow \min$$

- 1) Выбирают нулевое приближение $\mathbf{x}_0 = (x_{10}, x_{20}, \dots, x_{n0})$ и определяют область поиска минимума функции:

[illegible]

- 2) Фиксируют (считают постоянными) значения всех координат кроме x_1 . Тогда функция $f(\mathbf{x})$ будет зависеть только от одной переменной x_1 :

$$f_1(x_1) = f(x_1, x_{20} = \text{const}, \dots, x_{n0} = \text{const})$$

- 3) Используя метод одномерной минимизации, находится минимум функции

одной переменной $f_1(x_1) \rightarrow \min$, который можно обозначить через m_1 .

$$m_1 = \arg \left(\min_{a_1 \leq x_1 \leq b_1} f_1(x_1) \right)$$

- 4) Сделан переход из начальной точки x_0 в точку “частного” минимума по направлению, параллельному оси x_1 :

$$(x_{10}, x_{20}, \dots, x_{n0}) \rightarrow (m_1, x_{20}, \dots, x_{n0}),$$

и значение функции уменьшается:

$$f(x_{10}, x_{20}, \dots, x_{n0}) > f(m_1, x_{20}, \dots, x_{n0})$$

- 5) Затем из новой точки $(m_1, x_{20}, \dots, x_{n0})$ осуществляется спуск по направлению, параллельному оси x_2 , то есть находится минимум функции $f_2(x_2)$:

$$f_2(x_2) = f(m_1, x_2, x_{30} = \text{const}, \dots, x_{n0} = \text{const}),$$

который обозначим m_2 :

$$m_2 = \arg \left(\min_{a_2 \leq x_2 \leq b_2} f_2(x_2) \right)$$

- 6) Таким образом, сделан переход во вторую точку “частного” минимума по направлению, параллельному оси x_2 :

$$(m_1, x_{20}, \dots, x_{n0}) \rightarrow (m_1, m_2, \dots, x_{n0}),$$

и значение функции уменьшается:

$$f(m_1, x_{20}, \dots, x_{n0}) > f(m_1, m_2, \dots, x_{n0})$$

- 7) Процесс спуска по координатам повторяется для всех остальных переменных задачи x_3, x_4, \dots, x_n , а переход в точку $x_m = (m_1, m_2, \dots, m_n)$ завершает цикл спусков.
- 8) Конечную точку цикла спусков можно принять за нулевое приближение $x_0 = x_m$ и повторить цикл спусков по координатам x_1, x_2, \dots, x_n до тех пор, пока не выполнено условие останова итерационного процесса (рисунок 19).

Практически можно задать некоторое число $\epsilon > 0$, связанное с выбранной

точностью вычислений, и проводить итерации до тех пор, пока на k -ой итерации не будут выполнены одно или несколько неравенств вида:

$$\|\mathbf{x}_k - \mathbf{x}_{k-1}\| < \epsilon_1, \quad \|f(\mathbf{x}_k) - f(\mathbf{x}_{k-1})\| < \epsilon_2 \quad (13)$$

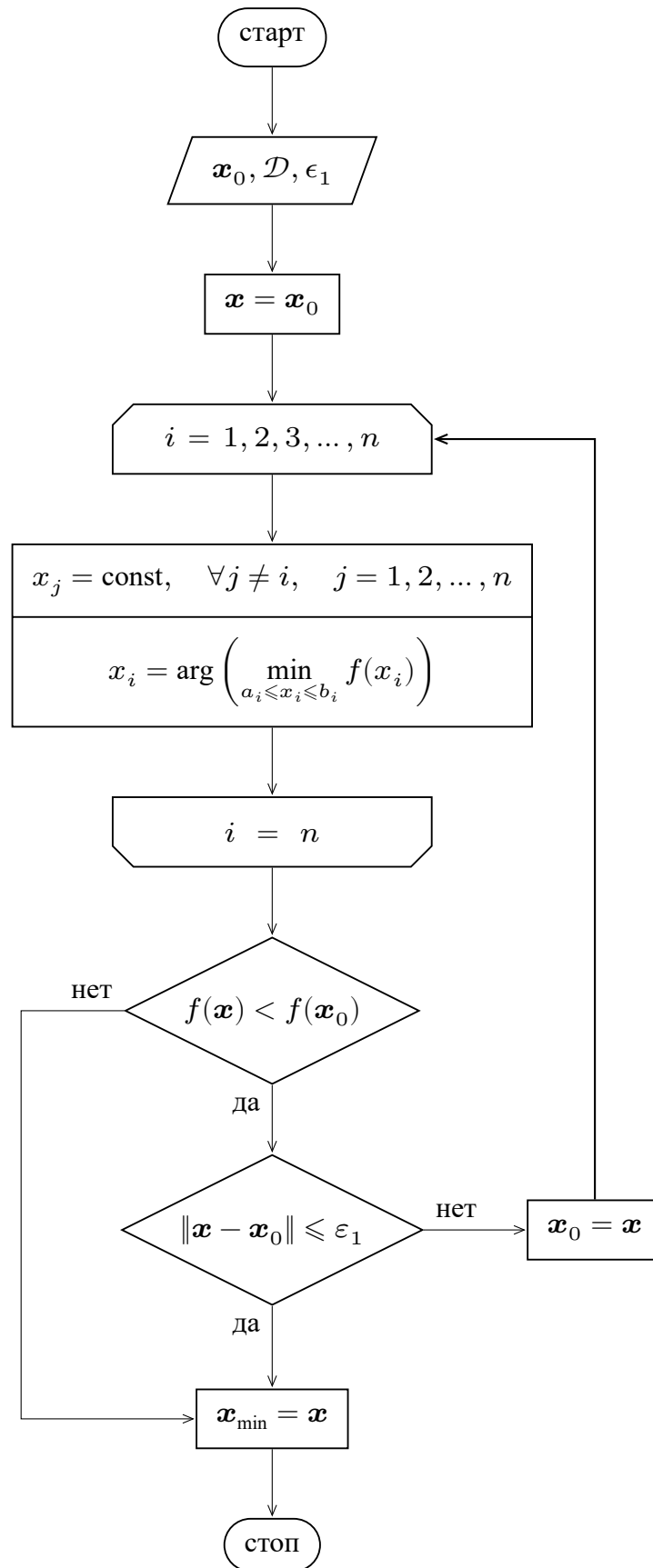


Рисунок 19 – Блок-схема алгоритма нахождения минимума функции $f(x)$ многих переменных методом по координатного спуска

6.2.2 Метод градиентного спуска

Градиентный спуск – метод нахождения локального экстремума (минимума или максимума) функции многих переменных $f(x_1, x_2, \dots, x_n)$ с помощью движения вдоль градиента этой функции. Это наиболее простой в реализации из всех методов локальной оптимизации, но имеет относительно малую (линейную) скорость сходимости.

Градиент ∇ это вектор, указывающий направление наибольшего возрастания некоторой функции f , значение которой меняется от одной точки пространства к другой (скалярного поля), а по величине (модулю) равный скорости роста этой величины в этом направлении. Компонентами вектора градиента являются частные производные f по всем её аргументам:

$$\nabla f = \left(\frac{\partial f}{\partial x_1}, \frac{\partial f}{\partial x_2}, \dots, \frac{\partial f}{\partial x_n} \right) \quad (14)$$

Для случая трёхмерного пространства градиентом скалярной функции $f(x, y, z)$ называется векторная функция:

$$\text{grad } f = \nabla f,$$

где ∇ – векторный дифференциальный оператор набла, компоненты которого являются частными производными по координатам:

$$\nabla = \left(\frac{\partial}{\partial x}, \frac{\partial}{\partial y}, \frac{\partial}{\partial z} \right)$$

Следует отметить, что оператор набла не принадлежит тому же пространству, что и обычные векторы, а говоря точнее, скалярное и векторное произведение для него определено с некоторыми отличиями. Оператор ∇ действует на те скалярные поля, что стоят от него справа, и не действует на стоящие от него слева. Поэтому скалярное и векторное произведение с участием ∇ *не коммутативны* и не антикоммутативны, как это свойственно для таких произведений обычных векторов.

Минимизация целевой функции $f(x)$ сводится к итерационному процессу последовательного выбора нового вектора неизвестных x_{k+1} , такого чтобы значение функции в новой точки было меньше чем в предыдущих:

$$f(x_0) > f(x_1) > \dots > f(x_k) > f(x_{k+1}) > \dots$$

Предполагая, что новый вектор неизвестных мало отличается от предыдущего ($\mathbf{x}_{k+1} - \mathbf{x}_k \approx 0$), можно воспользоваться линейным приближением для разложения в ряд Тейлора целевой функции:

$$f(\mathbf{x}_{k+1}) = f(\mathbf{x}_k) + (\nabla f(\mathbf{x}_k), \mathbf{x}_{k+1} - \mathbf{x}_k), \quad (15)$$

где k – номер итерационного шага процесса; \mathbf{x}_k – значение неизвестных на k -ой итерации.

Если в качестве нового вектора неизвестных выбрать:

$$\mathbf{x}_{k+1} = \mathbf{x}_k - \lambda \cdot \nabla f(\mathbf{x}_k), \quad (16)$$

то из (15) получим:

$$f(\mathbf{x}_{k+1}) = f(\mathbf{x}_k) - \lambda \cdot \|\nabla f(\mathbf{x}_k)\|^2 \rightarrow f(\mathbf{x}_{k+1}) \leq f(\mathbf{x}_k) \quad (17)$$

где $\lambda > 0$ – малое положительное число (параметр метода), имеющий смысл скорости градиентного спуска; $\|\nabla f(\mathbf{x}_k)\| \geq 0$ – норма вектора градиента (неотрицательное число):

$$\|\nabla f\| = \sqrt{(\nabla f, \nabla f)}$$

Таким образом, выбор нового вектора неизвестных \mathbf{x}_{k+1} в соответствии с выражением (16), гарантирует монотонное убывание целевой функции $f(\mathbf{x})$ в каждой итерации. Поэтому основная идея метода градиентного спуска заключается в том, чтобы последовательно идти в направлении наибольшего уменьшения целевой функции, которое задаётся антиградиентом $-\nabla f(\mathbf{x})$.

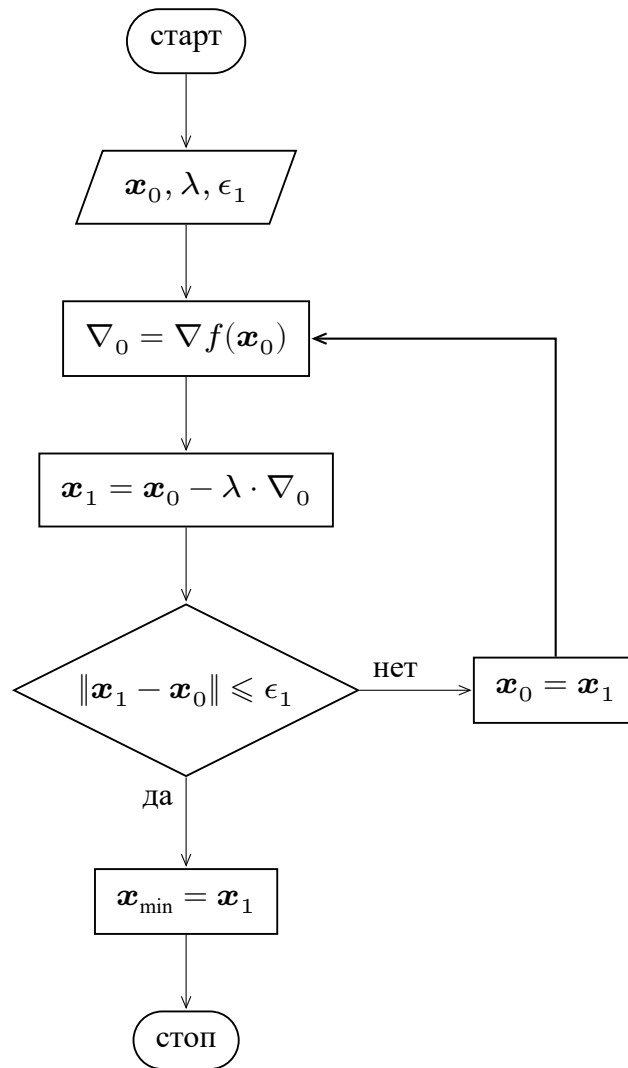


Рисунок 20 – Блок-схема алгоритма нахождения минимума функции $f(x)$ многих переменных методом градиентного спуска

6.2.3 Метод тяжелого шара

Поиск минимума функции многих переменных $f(x)$ методом “тяжелого шара” основан на аналогии движения материальной частицы массой m в консервативном силовом поле $F(x)$ в вязкой среде.

В соответствии с принципом минимальной энергии тело смещается в положение, которое минимизирует общую потенциальную энергию системы $f(x) \rightarrow \min$. Поэтому если предположить, что функция $f(x)$ является потенциальной энергией частицы в консервативном силовом поле $F(x) = -\nabla f(x)$, и частица перемещается в пространстве x минимизируя свою энергию, то урав-

нение движения этой частицы можно записать в виде:

$$\begin{cases} \frac{dx}{dt} = v \\ m \frac{dv}{dt} = F - \alpha \cdot v \end{cases} \quad (18)$$

где x – положение частицы в выбранной системе координат; v и α – скорость и коэффициент вязкого трения частицы в среде, соответственно.

Этот метод используется в методе стохастического градиентного спуска и в качестве расширения алгоритмов обратного распространения ошибок для обучения искусственных нейронных сетей.

Поиск минимума данным методом начинается с задания начальных условий, которые, как правило, формулируются в виде:

$$\begin{cases} x(0) = x_0 \\ v(0) = v_0 \end{cases}, \quad (19)$$

где x_0 – начальное приближения для поиска минимума функции; v_0 – “начальная скорость” в пространстве неизвестных.

Масса частицы m и коэффициент вязкого трения α являются эвристическими параметрами метода и выбираются произвольным образом, отражающим специфику решаемой задачи.

6.3 Численная оптимизация местоположения склада торговой сети

Рассмотрим задачу нахождения координат расположения склада готовой продукции (распределительного центра), таких чтобы суммарное расстояние от склада до потребителей продукции (магазинов торговой сети) было минимальным.

6.3.1 Метод по координатного спуска

Будем полагать, что магазины торговой сети находятся в городах, которые имеют определенные местоположения заданные их географическими координатами.

Географические координаты определяют местоположение точки на поверхности Земли, и строятся по принципу сферических координат. Центр системы географических координат помещается в центр Земли, а положение точки в пространстве определяется расстоянием r от центра Земли, широтой ϕ и

долготой λ (рисунок 21):

- 1) широта ϕ – угол между местным направлением зенита и плоскостью экватора, отсчитываемый в обе стороны от экватора.
- 2) долгота λ – двугранный угол между плоскостью меридиана, проходящего через данную точку, и плоскостью начального нулевого меридиана.

Радиус-вектор \mathbf{r} точки A , находящейся на поверхности Земли, имеет в декартовой системе координат компоненты:

$$\mathbf{r} = (r_x, r_y, r_z),$$

где $r_{x,y,z}$ – компоненты вектора \mathbf{r} , то есть проекции этого вектора на оси декартовой системы координат:

$$\begin{cases} r_x = \mathcal{R} \cdot \cos(\phi) \cdot \cos(\lambda) \\ r_y = \mathcal{R} \cdot \cos(\phi) \cdot \sin(\lambda) \\ r_z = \mathcal{R} \cdot \sin(\phi) \end{cases},$$

\mathcal{R} – радиус планеты Земля.

Таким образом, для определения местоположения *на поверхности Земли*, достаточно задать широту ϕ и долготу λ географического объекта.

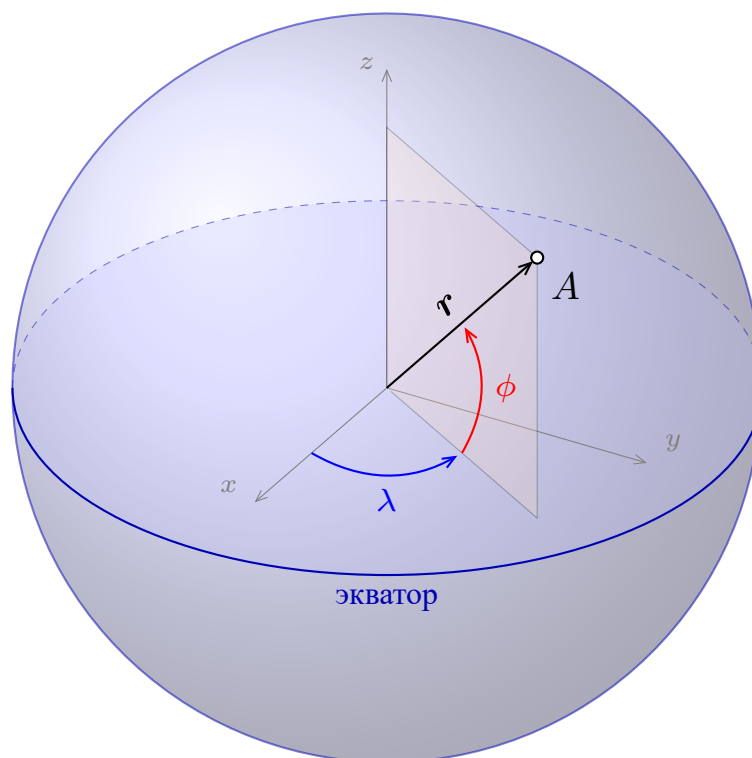


Рисунок 21 – Географическая система координат на поверхности Земли

Кратчайшее расстояние между двумя точками A и B находящимися на сферической поверхности Земли равно длине дуги большого круга. Если известны географические координаты точек $A(\phi_a, \lambda_a)$ и $B(\phi_b, \lambda_b)$, то расстояние на поверхности Земли между этими точками составляет (рисунок 22):

$$d_{AB} = \mathcal{R} \cdot \alpha, \quad (20)$$

где α – центральный угол между радиус-векторами \mathbf{a} и \mathbf{b} точек A и B , лежащих на поверхности Земли.

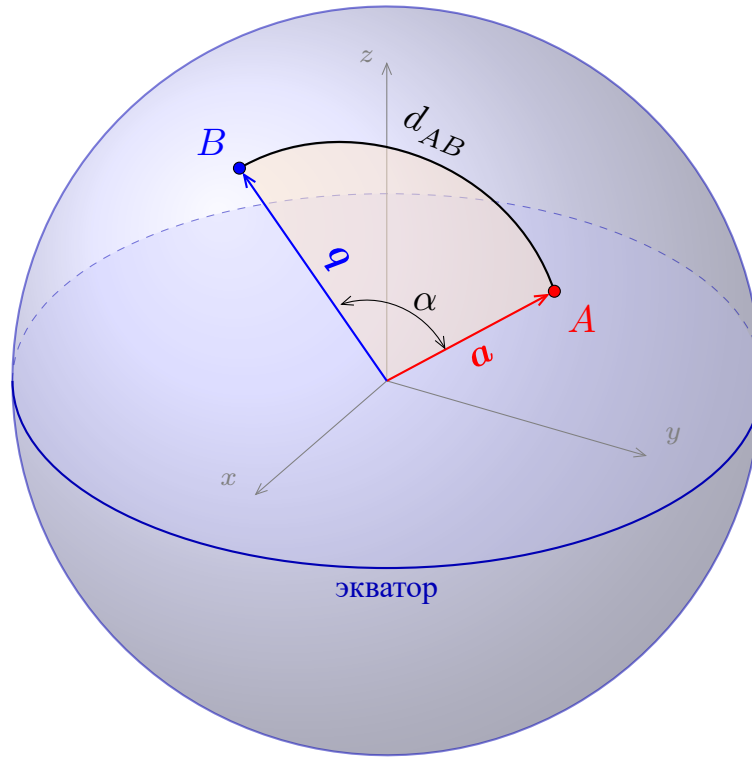


Рисунок 22 – Расстояние d_{AB} (длина дуги) на поверхности сферы между точками A и B

Косинус угла α между векторами \mathbf{a} и \mathbf{b} можно определить, воспользовавшись геометрическим свойством скалярного произведения этих векторов:

$$(\mathbf{a}, \mathbf{b}) = \|\mathbf{a}\| \cdot \|\mathbf{b}\| \cdot \cos(\alpha) \quad \rightarrow \quad \cos(\alpha) = \frac{(\mathbf{a}, \mathbf{b})}{\|\mathbf{a}\| \cdot \|\mathbf{b}\|}, \quad (21)$$

где $\|\mathbf{a}\| = \|\mathbf{b}\| = \mathcal{R}$ – длины радиус-векторов, с учетом постоянства радиуса Земли.

С алгебраической точки зрения, *скалярное произведение векторов* – это операция над двумя векторами \mathbf{a} и \mathbf{b} , результатом которой является число (скаляр), значение которого не зависит от выбранной системы координат, то есть скалярное произведение является *инвариантом*:

$$(\mathbf{a}, \mathbf{b}) = a_x \cdot b_x + a_y \cdot b_y + a_z \cdot b_z, \quad (22)$$

где $a_{x,y,z}$ и $b_{x,y,z}$ – компоненты векторов в декартовой системе координат:

$$\mathbf{a} = \begin{cases} a_x = \mathcal{R} \cdot \cos(\phi_a) \cdot \cos(\lambda_a) \\ a_y = \mathcal{R} \cdot \cos(\phi_a) \cdot \sin(\lambda_a) \\ a_z = \mathcal{R} \cdot \sin(\phi_a) \end{cases} \quad (23)$$

$$\mathbf{b} = \begin{cases} b_x = \mathcal{R} \cdot \cos(\phi_b) \cdot \cos(\lambda_b) \\ b_y = \mathcal{R} \cdot \cos(\phi_b) \cdot \sin(\lambda_b) \\ b_z = \mathcal{R} \cdot \sin(\phi_b) \end{cases} \quad (24)$$

Косинус угла (21) между векторами \mathbf{a} и \mathbf{b} с учетом (23) и (24):

$$\begin{aligned} \cos(\alpha) = & \cos(\phi_a) \cos(\lambda_a) \cdot \cos(\phi_b) \cos(\lambda_b) + \\ & \cos(\phi_a) \sin(\lambda_a) \cdot \cos(\phi_b) \sin(\lambda_b) + \\ & \sin(\phi_a) \cdot \sin(\phi_b) \end{aligned}$$

или

$$\cos(\alpha) = \cos(\phi_a) \cos(\phi_b) \cdot \left(\cos(\lambda_a) \cdot \cos(\lambda_b) + \sin(\lambda_a) \cdot \sin(\lambda_b) \right) + \sin(\phi_a) \sin(\phi_b)$$

Воспользовавшись формулой для разности углов тригонометрических функций (косинуса):

$$\cos(\lambda_a) \cdot \cos(\lambda_b) + \sin(\lambda_a) \cdot \sin(\lambda_b) = \cos(\lambda_a - \lambda_b)$$

окончательно получим:

$$\cos(\alpha) = \cos(\phi_a) \cos(\phi_b) \cdot \cos(\Delta\lambda) + \sin(\phi_a) \sin(\phi_b),$$

где $\Delta\lambda = \lambda_a - \lambda_b$ – разность долготы географических координат точек A и B , соответственно.

Таким образом, кратчайшее расстояние на поверхности Земли между двумя точками (20) может быть найдено из соотношения:

$$d_{AB} = \mathcal{R} \cdot \arccos(\cos(\phi_a) \cos(\phi_b) \cdot \cos(\Delta\lambda) + \sin(\phi_a) \sin(\phi_b)), \quad (25)$$

1) С помощью картографических веб сервисов (например, [Яндекс.Карты](#), [кар-](#)

ты Google, и т.п.). определим географические координаты всех городов, в которых находятся магазины торговой сети. Полученные данные представим в таблице 10.

Таблица 10 – Географические координаты магазинов торговой сети

| Город | Географические координаты | |
|---------|---------------------------|-------------------|
| | широта ϕ | долгота λ |
| Москва | 55.66352 | 37.62964 |
| Саратов | 51.53440 | 46.03121 |
| Воронеж | 56.67071 | 39.16190 |

- 2) Построим целевую функцию $f(\phi, \lambda)$ – суммарное расстояние вдоль поверхности Земли от склада до магазинов в городах Москва, Саратов и Воронеж:

$$f(\phi, \lambda) = d_1(\phi, \lambda) + d_2(\phi, \lambda) + d_3(\phi, \lambda),$$

где ϕ и λ – географическая широта и долгота местоположения склада; d_1 , d_2 и d_3 – расстояние вдоль поверхности Земли (25) между складом и городами Москва, Саратов и Воронеж, соответственно:

$$d_1(\phi, \lambda) = \mathcal{R} \cdot \arccos(\cos(\phi) \cos(\phi_1) \cdot \cos(\lambda - \lambda_1) + \sin(\phi) \sin(\phi_1))$$

$$d_2(\phi, \lambda) = \mathcal{R} \cdot \arccos(\cos(\phi) \cos(\phi_2) \cdot \cos(\lambda - \lambda_2) + \sin(\phi) \sin(\phi_2))$$

$$d_3(\phi, \lambda) = \mathcal{R} \cdot \arccos(\cos(\phi) \cos(\phi_3) \cdot \cos(\lambda - \lambda_3) + \sin(\phi) \sin(\phi_3))$$

где $\phi_1 = 55.66352$ и $\lambda_1 = 37.62964$ – широта и долгота города Москва; $\phi_2 = 51.53440$ и $\lambda_2 = 46.03121$ – широта и долгота города Саратов; $\phi_3 = 56.67071$ и $\lambda_3 = 39.16190$ – широта и долгота города Воронеж (таблица 10).

- 3) Для решения задачи оптимизации целевой функции *методом по координатного спуска* определим область поиска минимума функции $r(\phi, \lambda)$.

Например, в качестве границ области можно выбрать экстремальные значения широты и долготы городов, в которых расположены магазины торговой

сети:

$$\begin{cases} \phi_{\inf} = \min(\phi_1, \phi_2, \phi_3) = \phi_2 = 51.53440 \\ \phi_{\sup} = \max(\phi_1, \phi_2, \phi_3) = \phi_3 = 56.67071 \\ \lambda_{\inf} = \min(\lambda_1, \lambda_2, \lambda_3) = \lambda_1 = 37.62964 \\ \lambda_{\sup} = \max(\lambda_1, \lambda_2, \lambda_3) = \lambda_2 = 46.03121 \end{cases}$$

Таким образом, минимум целевой функции будем искать в области \mathcal{D} :

$$\mathcal{D} = \begin{cases} \phi_{\inf} \leq \phi \leq \phi_{\sup} \\ \lambda_{\inf} \leq \lambda \leq \lambda_{\sup} \end{cases} = \begin{cases} 51.53440 \leq \phi \leq 56.67071 \\ 37.62964 \leq \lambda \leq 46.03121 \end{cases}$$

- 4) Критерием прекращения итерационного поиска минимума целевой функции считать условие выполнения одного из неравенств:

$$\epsilon_1 \geq \xi, \quad f(\phi_k, \lambda_k) \geq f(\phi_{k-1}, \lambda_{k-1}) \quad (26)$$

где $\epsilon_1 = 0.1$ – заданная точность определения местоположения склада;
 ξ – изменение местоположения склада на текущем k шаге итерационного процесса:

$$\xi = \sqrt{(\phi_k - \phi_{k-1})^2 + (\lambda_k - \lambda_{k-1})^2}; \quad (27)$$

ϕ_k и λ_k – значение географических координат склада на k -ом шаге итерационного процесса.

- 5) В качестве начального местоположение склада (ϕ_0, λ_0) выберем любую произвольную точку внутри области поиска минимума целевой функции, например:

$$\begin{cases} \phi_0 = 52.0 \\ \lambda_0 = 44.0 \end{cases}.$$

- 6) Считаем постоянными значения долготы склада $\lambda = \lambda_0 = \text{const}$ и рассматриваем вспомогательную функцию только одной переменной ϕ :

$$f_1(\phi) = f(\phi, \lambda_0) = f(\phi, 44.0)$$

Находим минимум вспомогательной функции одной переменной $f_1(\phi)$, ис-

пользуя метод одномерной минимизации:

$$\phi_{\min} = \arg \left(\min_{\phi_{\inf} \leq \phi \leq \phi_{\sup}} f_1(\phi) \right) = 54.66658$$

где ϕ_{\min} – положение “частного” минимума функции $f(\phi, 44.0)$.

Считаем постоянными значения широты склада $\phi = \phi_{\min} = \text{const}$ и рассматриваем вторую вспомогательную функцию только одной переменной λ :

$$f_2(\lambda) = f(\phi_{\min}, \lambda) = f(54.66658, \lambda)$$

Находим минимум функции одной переменной $f_2(\lambda)$, используя метод одномерной минимизации:

$$\lambda_{\min} = \arg \left(\min_{\lambda_{\inf} \leq \lambda \leq \lambda_{\sup}} f_2(\lambda) \right) = 39.49103$$

где λ_{\min} – положение “частного” минимума функции $f(54.66658, \lambda)$.

- 7) Определяем изменение местоположения склада (27) в результате проведенных итерационных процессов (рисунок 23):

$$\begin{aligned} \xi &= \sqrt{(\phi_{\min} - \phi_0)^2 + (\lambda_{\min} - \lambda_0)^2} = \\ &= \sqrt{(54.66658 - 52.0)^2 + (39.49103 - 44.0)^2} = \\ &= 5.23846 \end{aligned}$$

Находим значение целевой функции в начальной $f(\phi_0, \lambda_0)$ и конечной точки $f(\phi_{\min}, \lambda_{\min})$ точках местоположения склада:

$$\begin{aligned} f(\phi_0, \lambda_0) &= f(52.0, 44.0) &= 1338.696 \\ f(\phi_{\min}, \lambda_{\min}) &= f(54.66658, 39.49103) &= 944.004 \end{aligned}$$

Сравниваем величину текущего “шага” ξ и заданную точность расчетов ϵ_1 , а также значения целевой функции $f(\phi, \lambda)$ в начальной и конечной точки итерационного процесса:

$$\begin{aligned} \epsilon_1 = 0.1 &< \xi = 5.23846 \\ f(\phi_{\min}, \lambda_{\min}) = 944.004 &< f(\phi_0, \lambda_0) = 1338.696 \end{aligned}$$

Критерий остановки итерационного процесса (26) не выполняется, следова-

тельно, итерационный процесс продолжаем!

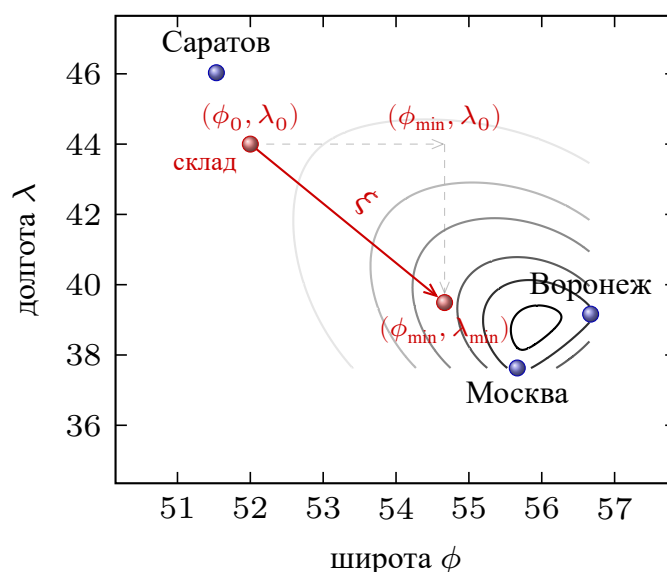


Рисунок 23 – “Рельеф” целевой функции $f(\phi, \lambda)$ и “траектория” поиска местоположения склада (ϕ, λ) торговой сети (маркерами обозначены города)

- 8) За новое местоположение склада торговой сети принимается положение “частного” минимума целевой функции на предыдущем шаге итерационного процесса:

$$\begin{cases} \phi_0 = \phi_{\min} = 54.66658 \\ \lambda_0 = \lambda_{\min} = 39.49103 \end{cases}$$

- 9) Считаем постоянными значения долготы склада $\lambda = \lambda_0 = \text{const}$ и рассматриваем вспомогательную функцию только одной переменной ϕ :

$$f_1(\phi) = f(\phi, \lambda_0) = f(\phi, 39.49103)$$

Находим минимум вспомогательной функции одной переменной $f_1(\phi)$, используя метод одномерной минимизации:

$$\phi_{\min} = \arg \left(\min_{\phi_{\inf} \leq \phi \leq \phi_{\sup}} f_1(\phi) \right) = 55.94734$$

где ϕ_{\min} – положение “частного” минимума функции $f(\phi, 39.49103)$.

Считаем постоянными значения широты склада $\phi = \phi_{\min} = \text{const}$ и рассмат-

риваем вторую вспомогательную функцию только одной переменной λ :

$$f_2(\lambda) = f(\phi_{\min}, \lambda) = f(55.94734, \lambda)$$

Находим минимум функции одной переменной $f_2(\lambda)$, используя метод одномерной минимизации:

$$\lambda_{\min} = \arg \left(\min_{\lambda_{\inf} \leq \lambda \leq \lambda_{\sup}} f_2(\lambda) \right) = 38.88818$$

где λ_{\min} – положение “частного” минимума функции $f(55.94734, \lambda)$.

10) Определяем изменение местоположения склада (27) в результате проведенных итерационных процессов (рисунок 24):

$$\begin{aligned} \xi &= \sqrt{(\phi_{\min} - \phi_0)^2 + (\lambda_{\min} - \lambda_0)^2} = \\ &= \sqrt{(55.94734 - 54.66658)^2 + (38.88818 - 39.49103)^2} = \\ &= 1.41555 \end{aligned}$$

Находим значение целевой функции в начальной $f(\phi_0, \lambda_0)$ и конечной точки $f(\phi_{\min}, \lambda_{\min})$ точках местоположения склада:

$$\begin{aligned} f(\phi_0, \lambda_0) &= f(54.66658, 39.49103) = 944.004 \\ f(\phi_{\min}, \lambda_{\min}) &= f(55.94734, 38.88818) = 845.572 \end{aligned}$$

Сравниваем величину текущего “шага” ξ и заданную точность расчетов ϵ_1 , а также значения целевой функции $f(\phi, \lambda)$ в начальной и конечной точки итерационного процесса:

$$\begin{aligned} \epsilon_1 = 0.1 &< \xi = 1.41555 \\ f(\phi_{\min}, \lambda_{\min}) = 845.572 &< f(\phi_0, \lambda_0) = 944.004 \end{aligned}$$

Критерий остановки итерационного процесса (26) не выполняется, следовательно, *итерационный процесс продолжаем!*

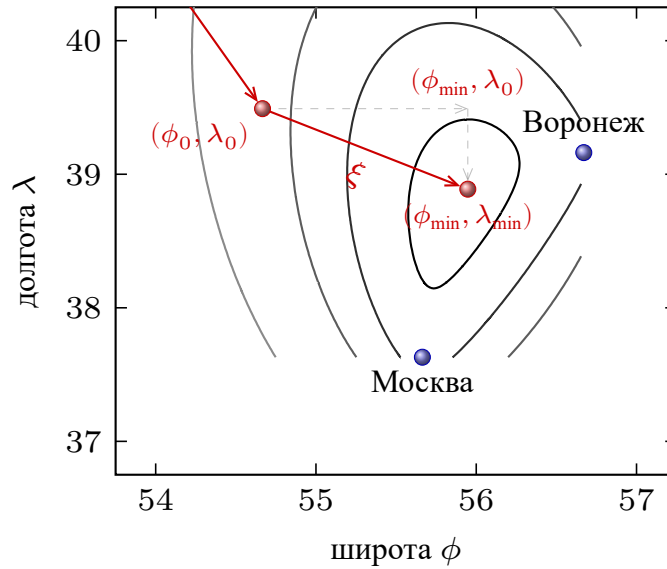


Рисунок 24 – “Рельеф” целевой функции $f(\phi, \lambda)$ и “траектория” поиска местоположения склада (ϕ, λ) торговой сети (маркерами обозначены города)

- 11) За новое местоположение склада торговой сети принимается положение “частного” минимума целевой функции на предыдущем шаге итерационного процесса:

$$\begin{cases} \phi_0 = \phi_{\min} = 55.94734 \\ \lambda_0 = \lambda_{\min} = 38.88818 \end{cases}$$

- 12) Считаем постоянными значения долготы склада $\lambda = \lambda_0 = \text{const}$ и рассматриваем вспомогательную функцию только одной переменной ϕ :

$$f_1(\phi) = f(\phi, \lambda_0) = f(\phi, 38.88818)$$

Находим минимум вспомогательной функции одной переменной $f_1(\phi)$, используя метод одномерной минимизации:

$$\phi_{\min} = \arg \left(\min_{\phi_{\inf} \leq \phi \leq \phi_{\sup}} f_1(\phi) \right) = 55.88973$$

где ϕ_{\min} – положение “частного” минимума функции $f(\phi, 38.88818)$.

Считаем постоянными значения широты склада $\phi = \phi_{\min} = \text{const}$ и рассматриваем вторую вспомогательную функцию только одной переменной λ :

$$f_2(\lambda) = f(\phi_{\min}, \lambda) = f(55.88973, \lambda)$$

Находим минимум функции одной переменной $f_2(\lambda)$, используя метод одномерной минимизации:

$$\lambda_{\min} = \arg \left(\min_{\lambda_{\inf} \leq \lambda \leq \lambda_{\sup}} f_2(\lambda) \right) = 38.83941$$

где λ_{\min} – положение “частного” минимума функции $f(55.88973, \lambda)$.

- 13) Определяем изменение местоположения склада (27) в результате проведенных итерационных процессов (рисунок 25):

$$\begin{aligned} \xi &= \sqrt{(\phi_{\min} - \phi_0)^2 + (\lambda_{\min} - \lambda_0)^2} = \\ &= \sqrt{(55.88973 - 55.94734)^2 + (38.83941 - 38.88818)^2} = \\ &= 0.07548 \end{aligned}$$

Находим значение целевой функции в начальной $f(\phi_0, \lambda_0)$ и конечной точки $f(\phi_{\min}, \lambda_{\min})$ точках местоположения склада:

$$\begin{aligned} f(\phi_0, \lambda_0) &= f(55.94734, 38.88818) = 845.572 \\ f(\phi_{\min}, \lambda_{\min}) &= f(55.88973, 38.83941) = 845.314 \end{aligned}$$

Сравниваем величину текущего “шага” ξ и заданную точность расчетов ϵ_1 , а также значения целевой функции $f(\phi, \lambda)$ в начальной и конечной точки итерационного процесса:

$$\begin{aligned} \epsilon_1 = 0.1 &> \xi = 0.07548 \\ f(\phi_{\min}, \lambda_{\min}) &= 845.314 < f(\phi_0, \lambda_0) = 845.572 \end{aligned}$$

Критерий остановки итерационного процесса (26) выполняется, следовательно, *итерационный процесс прекращаем!*

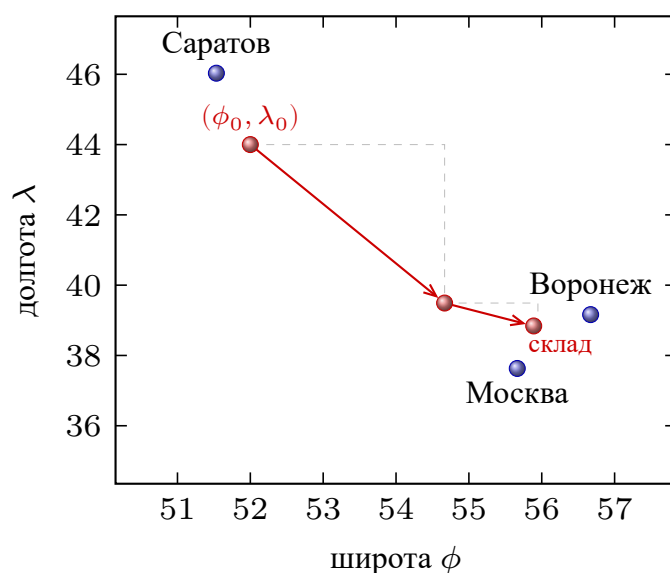


Рисунок 25 – “Рельеф” целевой функции $f(\phi, \lambda)$ и “траектория” поиска местоположения склада (ϕ, λ) торговой сети (маркерами обозначены города)

14) Полученные данные в процессе выполнения минимизации целевой функции методом по координатного спуска представлены в таблице 11.

Таблица 11 – “Траектория” поиска местоположения склада торговой сети

| Итерация | Местоположение склада | | Шаг ξ | Расстояние $f(\phi, \lambda)$, км |
|----------|-----------------------|-------------------|-----------|------------------------------------|
| | широта ϕ | долгота λ | | |
| 0 | 52.00000 | 44.00000 | 0.00000 | 1338.696 |
| 1 | 54.66658 | 39.49103 | 5.23846 | 944.004 |
| 2 | 55.94734 | 38.88818 | 1.41555 | 845.572 |
| 3 | 55.88973 | 38.83941 | 0.07548 | 845.314 |

На рисунке 26 представлена зависимость суммарного расстояния между складом и всеми магазинами от номера итерации k поиска минимума целевой функции:

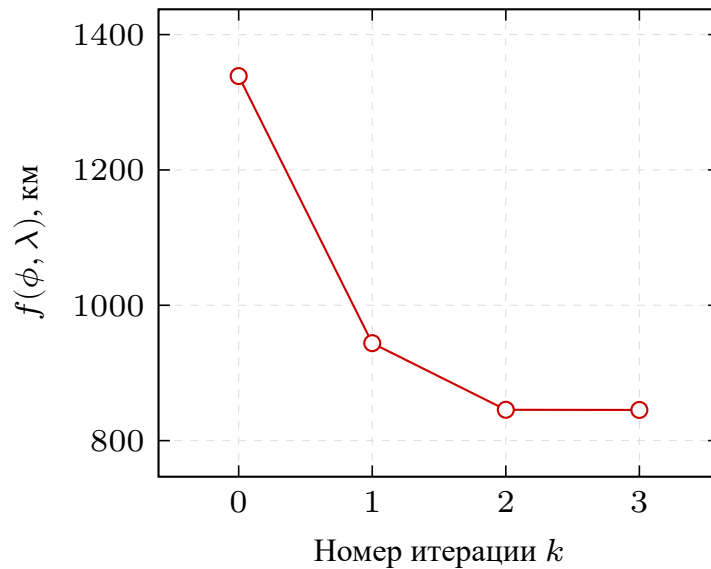


Рисунок 26 – Зависимость суммарного расстояния $f(\phi, \lambda)$ между складом и магазинами торговой сети от числа итераций

На рисунке представлена зависимость изменение местоположения склада ξ от номера итерации k процесса поиска минимума целевой функции:

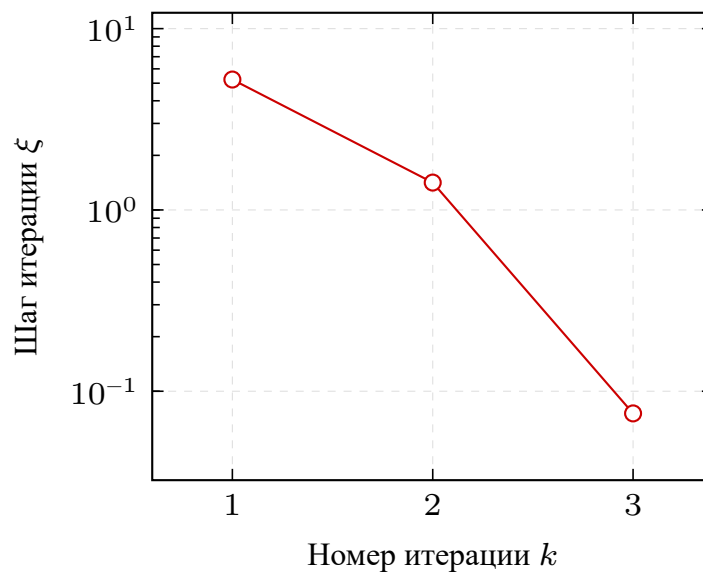
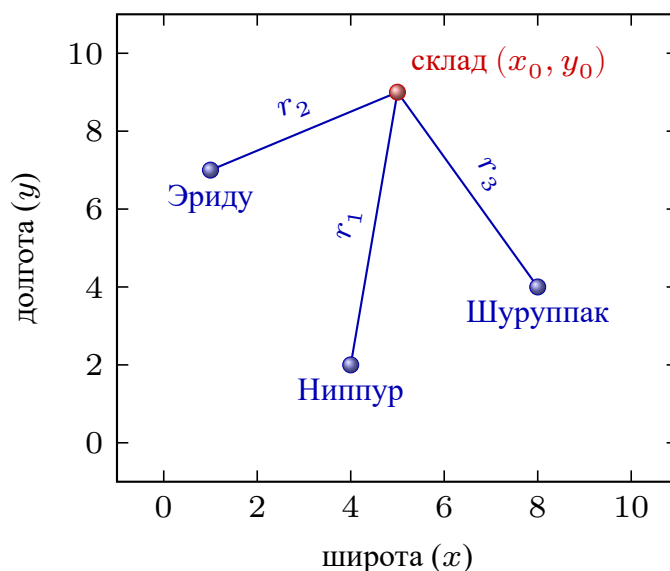


Рисунок 27 – Изменение местоположения склада ξ на k -ом шаге итерационного процесса

6.3.2 Метод градиентного спуска

Известны местоположение трех городов торговой сети в декартовой системе координат: **Ниппур** (4, 2), **Эриду** (1, 7) **Шуруппак** (8, 4).



1) Обозначим неизвестные:

x и y – положения склада по горизонтальной и вертикальной оси координат, соответственно.

2) Целевая функция – суммарное расстояние от склада до всех магазинов:

$$f = r_1 + r_2 + r_3,$$

где r_1 , r_2 и r_3 – расстояние от *склада* до городов Ниппур, Эриду и Шуруппак, соответственно.

3) В выбранной декартовой системе координат и для нахождения расстояния от склада до каждого города воспользуемся теоремой **Пифагора Самосского**:

$$\begin{aligned} r_1 &= \sqrt{(x_1 - x)^2 + (y_1 - y)^2} \\ r_2 &= \sqrt{(x_2 - x)^2 + (y_2 - y)^2} \\ r_3 &= \sqrt{(x_3 - x)^2 + (y_3 - y)^2}, \end{aligned}$$

где x_1 и y_1 – декартовы координаты города Ниппур; x_2 и y_2 – декартовы координаты города Эриду; x_3 и y_3 – декартовы координаты города Шуруппак. Таким образом, целевая функция – суммарное расстояние от склада до всех городов, с учетом данных задания о координатах городов (Ниппур, Эриду и

Шуруппак), запишется в виде:

$$f(x, y) = \sqrt{(4-x)^2 + (2-y)^2} + \sqrt{(1-x)^2 + (7-y)^2} + \sqrt{(8-x)^2 + (4-y)^2}$$

4) Определим градиент целевой функции $\nabla f(x, y)$:

$$\nabla f(x, y) = \left(\frac{\partial f}{\partial x}, \frac{\partial f}{\partial y} \right)$$

Для этого найдем частные производные целевой функции от координат x и y положения склада:

$$\begin{aligned} \frac{\partial f}{\partial x} = & -\frac{4-x}{\sqrt{(4-x)^2 + (2-y)^2}} \\ & -\frac{1-x}{\sqrt{(1-x)^2 + (7-y)^2}} \\ & -\frac{8-x}{\sqrt{(8-x)^2 + (4-y)^2}} \end{aligned}$$

$$\begin{aligned} \frac{\partial f}{\partial y} = & -\frac{2-y}{\sqrt{(4-x)^2 + (2-y)^2}} \\ & -\frac{7-y}{\sqrt{(1-x)^2 + (7-y)^2}} \\ & -\frac{4-y}{\sqrt{(8-x)^2 + (4-y)^2}} \end{aligned}$$

5) Выбираем (в общем случае, произвольно) начальные координаты склада, например, $x_0 = 5$ и $y_0 = 9$, скорость градиентного спуска $\lambda = 2$ и точность расчёта $\epsilon_1 = 0.25$ (единиц измерения).

6) Текущее суммарное расстояние от склада до всех городов:

$$\begin{aligned} R_0 = f(5, 9) = & \sqrt{(4-5)^2 + (2-9)^2} \\ & + \sqrt{(1-5)^2 + (7-9)^2} \\ & + \sqrt{(8-5)^2 + (4-9)^2} = 17.37 \end{aligned}$$

Определим градиент целевой функции в начальной точке положения склада

(x_0, y_0) :

$$\begin{aligned} \left. \frac{\partial f}{\partial x} \right|_{\substack{x=5 \\ y=9}} &= -\frac{4-5}{\sqrt{(4-5)^2 + (2-9)^2}} \\ &\quad -\frac{1-5}{\sqrt{(1-5)^2 + (7-9)^2}} \\ &\quad -\frac{8-5}{\sqrt{(8-5)^2 + (4-9)^2}} = 0.52 \end{aligned}$$

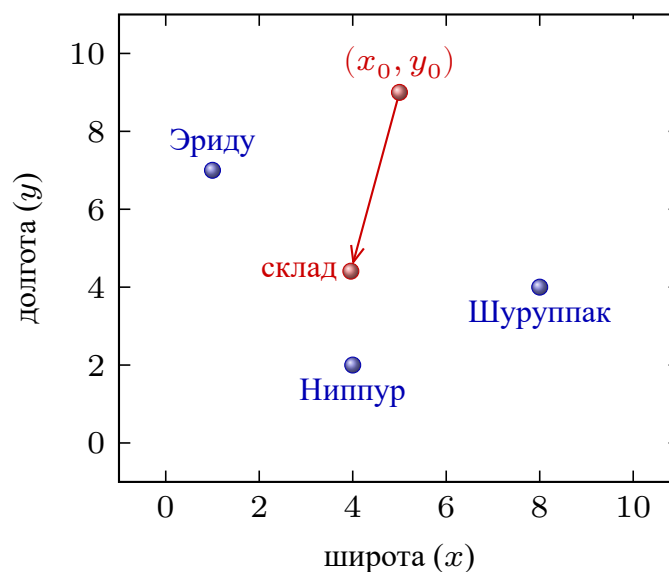
$$\begin{aligned} \left. \frac{\partial f}{\partial y} \right|_{\substack{x=5 \\ y=9}} &= -\frac{2-9}{\sqrt{(4-5)^2 + (2-9)^2}} \\ &\quad -\frac{7-9}{\sqrt{(1-5)^2 + (7-9)^2}} \\ &\quad -\frac{4-9}{\sqrt{(8-5)^2 + (4-9)^2}} = 2.29 \end{aligned}$$

Зная градиент целевой функции в начальной точке $\nabla f(x_0, y_0) = (0.52, 2.29)$, определим новые координаты склада:

$$x_1 = x_0 - \lambda \cdot \left. \frac{\partial f}{\partial x} \right|_{\substack{x=5 \\ y=9}} = 5 - 2 \cdot 0.52 = 3.96$$

$$y_1 = y_0 - \lambda \cdot \left. \frac{\partial f}{\partial y} \right|_{\substack{x=5 \\ y=9}} = 9 - 2 \cdot 2.29 = 4.41$$

Новые координаты склада $(x_1 = 3.96; y_1 = 4.41)$.



Рассчитаем величину “шага” – расстояния между двумя последовательными положениями склада:

$$\xi = \sqrt{(5 - 3.96)^2 + (9 - 4.41)^2} = 4.71$$

Сравниваем величину текущего “шага” ξ и заданную точность расчетов ϵ_1 :

$$\xi = 4.71 > \epsilon_1 = 0.25$$

Величина текущего “шага” r больше заданной точности расчетов ϵ_1 , следовательно, *итерационный процесс продолжаем!*

7) Текущее суммарное расстояние от склада до всех городов:

$$\begin{aligned} R_1 = f(3.96, 4.41) = & \sqrt{(4 - 3.96)^2 + (2 - 4.41)^2} \\ & + \sqrt{(1 - 3.96)^2 + (7 - 4.41)^2} \\ & + \sqrt{(8 - 3.96)^2 + (4 - 4.41)^2} = 10.41 \end{aligned}$$

Суммарное расстояние уменьшилось:

$$R_1 = 10.41 < R_0 = 17.37.$$

Рассчитаем градиент целевой функции в новой точке положения склада $x_1 = 3.96$ и $y_1 = 4.41$:

$$\begin{aligned} \left. \frac{\partial f}{\partial x} \right|_{\substack{x_1 = 3.96 \\ y_1 = 4.41}} = & -\frac{4 - 3.96}{\sqrt{(4 - 3.96)^2 + (2 - 4.41)^2}} \\ & -\frac{1 - 3.96}{\sqrt{(1 - 3.96)^2 + (7 - 4.41)^2}} \\ & -\frac{8 - 3.96}{\sqrt{(8 - 3.96)^2 + (4 - 4.41)^2}} = -0.26 \end{aligned}$$

$$\begin{aligned} \left. \frac{\partial f}{\partial y} \right|_{\substack{x_1 = 3.96 \\ y_1 = 4.41}} = & -\frac{2 - 4.41}{\sqrt{(4 - 3.96)^2 + (2 - 4.41)^2}} \\ & -\frac{7 - 4.41}{\sqrt{(1 - 3.96)^2 + (7 - 4.41)^2}} \\ & -\frac{4 - 4.41}{\sqrt{(8 - 3.96)^2 + (4 - 4.41)^2}} = 0.44 \end{aligned}$$

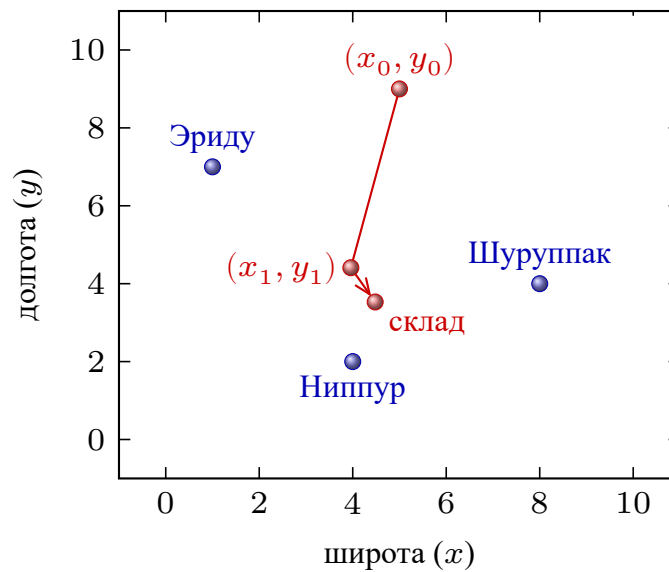
Зная градиент целевой функции в текущей точке $\nabla f(x_1, y_1) = (-0.26, 0.44)$,

определим новые координаты склада:

$$x_2 = x_1 - \lambda \cdot \left. \frac{\partial f}{\partial x} \right|_{\substack{x=3.96 \\ y=4.41}} = 3.96 - 2 \cdot (-0.26) = 4.48$$

$$y_2 = y_1 - \lambda \cdot \left. \frac{\partial f}{\partial y} \right|_{\substack{x=3.96 \\ y=4.41}} = 4.41 - 2 \cdot 0.44 = 3.53$$

Новые координаты склада ($x_2 = 4.48, y_2 = 3.53$).



Рассчитаем величину “шага” – расстояния между двумя последовательными положениями склада:

$$\xi = \sqrt{(3.96 - 4.48)^2 + (4.41 - 3.53)^2} = 1.03$$

Сравниваем величину текущего “шага” ξ и заданную точность расчетов ϵ_1 :

$$\xi = 1.03 > \epsilon_1 = 0.25$$

Величина текущего “шага” ξ больше заданной точности расчетов ϵ_1 , следовательно, *итерационный процесс продолжаем!*

8) Текущее суммарное расстояние от склада до всех городов:

$$R_2 = f(4.48, 3.53) = \sqrt{(4 - 4.48)^2 + (2 - 3.53)^2} + \sqrt{(1 - 4.48)^2 + (7 - 3.53)^2} + \sqrt{(8 - 4.48)^2 + (4 - 3.53)^2} = 10.07$$

Рассчитаем градиент целевой функции в новой точке положения склада $x_2 = 4.48$ и $y_2 = 3.53$:

$$\left. \frac{\partial f}{\partial x} \right|_{\substack{x_2 = 4.48 \\ y_2 = 3.53}} = -\frac{4 - 4.48}{\sqrt{(4 - 4.48)^2 + (2 - 3.53)^2}} - \frac{1 - 4.48}{\sqrt{(1 - 4.48)^2 + (7 - 3.53)^2}} - \frac{8 - 4.48}{\sqrt{(8 - 4.48)^2 + (4 - 3.53)^2}} = 0.02$$

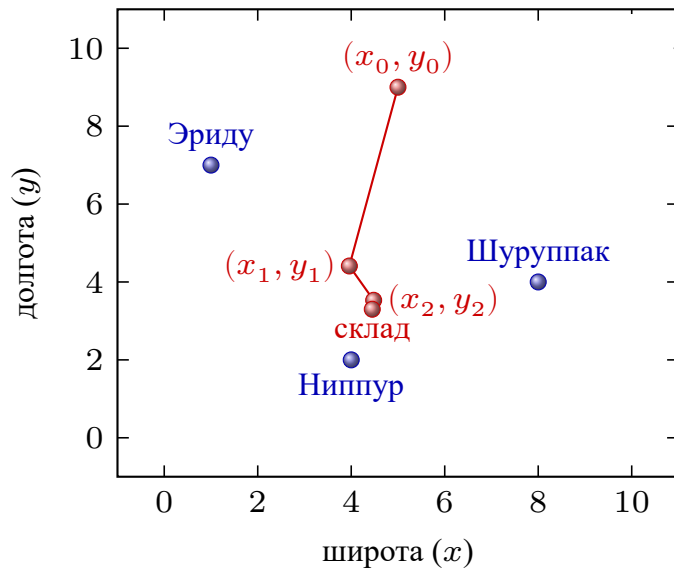
$$\left. \frac{\partial f}{\partial y} \right|_{\substack{x_2 = 4.48 \\ y_2 = 3.53}} = -\frac{2 - 3.53}{\sqrt{(4 - 4.48)^2 + (2 - 3.53)^2}} - \frac{7 - 3.53}{\sqrt{(1 - 4.48)^2 + (7 - 3.53)^2}} - \frac{4 - 3.53}{\sqrt{(8 - 4.48)^2 + (4 - 3.53)^2}} = 0.11$$

Зная градиент целевой функции в текущей точке $\nabla f(x_2, y_2) = (0.02, 0.11)$, определяют новое географическое положение склада:

$$x_3 = x_2 - \lambda \cdot \left. \frac{\partial f}{\partial x} \right|_{\substack{x=4.48 \\ y=3.53}} = 4.48 - 2 \cdot 0.02 = 4.45$$

$$y_3 = y_2 - \lambda \cdot \left. \frac{\partial f}{\partial y} \right|_{\substack{x=4.48 \\ y=3.53}} = 3.53 - 2 \cdot 0.11 = 3.30$$

Новые координаты склада ($x_0 = 4.45, y_0 = 3.30$).



Рассчитаем величину “шага” ξ – расстояния между двумя последовательными положениями склада:

$$\xi = \sqrt{(4.48 - 4.45)^2 + (3.53 - 3.30)^2} = 0.23$$

Сравниваем величину текущего “шага” ξ и заданную точность расчетов ϵ_1 :

$$\xi = 0.23 < \epsilon_1 = 0.25$$

Величина текущего “шага” ξ меньше заданной точности расчетов ϵ_1 , поэтому итерационный процесс поиска положения склада *прекращаем!*

9) Определим минимальное расстояние от склада до всех городов:

$$\begin{aligned} R_4 = f(4.45, 3.30) = & \sqrt{(4 - 4.45)^2 + (2 - 3.30)^2} \\ & + \sqrt{(1 - 4.45)^2 + (7 - 3.30)^2} \\ & + \sqrt{(8 - 4.45)^2 + (4 - 3.30)^2} = 10.05 \end{aligned}$$

Таким образом, с заданной точностью определены оптимальные координаты склада $(x_{\min} = 4.45, y_{\min} = 3.30)$, при которых общее расстояние от всех городов до склада будет минимальным и составит $R_{\min} = 10.05$:

7 Аппроксимация функция

Задача о приближении функции ставится следующим образом: данную функцию $f(x)$ необходимо заменить обобщенным полиномом $p_m(x)$ заданного порядка m так, чтобы отклонение (в известном смысле) функции $f(x)$ от обобщенного полинома $p_m(x)$ на указанном множестве $x = \{x\}$ было наименьшим. При этом полином $p_m(x)$ в общем случае называется аппроксимирующим.

Если множество x состоит из отдельных точек $x \in \{x_0, x_1, x_2, \dots, x_n\}$ (узлов), то приближение называется *точечным*. Если x есть отрезок $x_a < x < x_b$, то приближение называется *интегральным*. Для практики важным является приближение функций алгебраическими и тригонометрическими полиномами.

В случае постановки задачи поиска аппроксимирующей функции, которая обеспечивает погрешность не хуже заданной, необходимо подбирать и структуру этой функции. Эта задача значительно сложнее предыдущей и подходы в её решении основываются на переборе различных функций $p_m(x)$ и сравнении мер близости результатов расчета с исходными данными.

7.1 Точечное квадратичное аппроксимирование функций

На практике часто бывает, что заданный порядок m приближающего полинома $p_m(x)$ меньше числа узлов аппроксимации $m < n$, в которых известно значение функции $y_i = f(x_i)$ ($i = 0, 1, 2, \dots, n$). В этом случае обычно используют точечный метод наименьших квадратов и рассматривается алгебраический полином степени m вида:

$$p_m(x) = c_0 + c_1 \cdot x + c_2 \cdot x^2 + \dots + c_m \cdot x^m = \sum_{j=0}^m c_j \cdot x^j. \quad (28)$$

В качестве меры отклонения $\|r\|$ полинома $p_m(x)$ от известной функции $y(x)$ на множестве точек $\{x_0, x_1, x_2, \dots, x_n\}$, как правило, принимается сумма квадратов отклонений полинома от этой функции на заданной системе точек:

$$\|r\| = \sum_{i=0}^n (p_m(x_i) - y_i)^2 \quad (29)$$

Следует отметить, что мера отклонения полинома от известной функции есть функция многих переменных $\|r\| = \rho(c_0, c_1, \dots, c_m)$, т.е. коэффициентов полинома c_i ($i = 0, 1, \dots, m$), которые необходимо подобрать так, чтобы вели-

чина меры отклонения была наименьшей $\|r\| \rightarrow \min$. Полученный полином называется аппроксимирующим для данной функции, а процесс построения этого полинома – точечной квадратичной аппроксимацией или точечным квадратичным аппроксимированием функции.

Для решения задачи точечного квадратичного аппроксимирования, т.е. определения числовых значений всех коэффициентов полинома $p_m(x)$, необходимо найти *положения минимума функции* многих переменных $\rho(c_0, c_1, \dots, c_m)$.

Определим частные производные от величины суммы квадратов отклонений и воспользовавшись условием экстремума функции многих переменных, составим систему уравнений вида:

$$\frac{\partial \rho}{\partial c_0} = \frac{\partial \rho}{\partial c_1} = \frac{\partial \rho}{\partial c_2} = \dots = \frac{\partial \rho}{\partial c_m} = 0$$

Для определения неизвестных коэффициентов полинома $c_0, c_1, c_2, \dots, c_m$ необходимо решить систему $m + 1$ уравнений с $m + 1$ неизвестными:

$$\left\{ \begin{array}{l} \frac{\partial \rho}{\partial c_0} = 2 \cdot \sum_{i=0}^n (c_0 + c_1 \cdot x_i + c_2 \cdot x_i^2 + \dots + c_m \cdot x_i^m - y_i) \cdot 1 = 0 \\ \frac{\partial \rho}{\partial c_1} = 2 \cdot \sum_{i=0}^n (c_0 + c_1 \cdot x_i + c_2 \cdot x_i^2 + \dots + c_m \cdot x_i^m - y_i) \cdot x_i = 0 \\ \frac{\partial \rho}{\partial c_2} = 2 \cdot \sum_{i=0}^n (c_0 + c_1 \cdot x_i + c_2 \cdot x_i^2 + \dots + c_m \cdot x_i^m - y_i) \cdot x_i^2 = 0 \\ \dots = \dots = 0 \\ \frac{\partial \rho}{\partial c_m} = 2 \cdot \sum_{i=0}^n (c_0 + c_1 \cdot x_i + c_2 \cdot x_i^2 + \dots + c_m \cdot x_i^m - y_i) \cdot x_i^m = 0 \end{array} \right.$$

Таким образом, задача точечной квадратичной аппроксимации функции сводится к решению системы линейных уравнений относительно неизвестных – коэффициентов полинома $\{c_0, c_1, c_2, \dots, c_m\}$:

$$\mathbf{A} \cdot \mathbf{c} = \mathbf{b} \quad \text{или} \quad \begin{pmatrix} a_{00} & a_{01} & \dots & a_{0m} \\ a_{10} & a_{11} & \dots & a_{1m} \\ \vdots & \vdots & \ddots & \vdots \\ a_{m0} & a_{m1} & \dots & a_{mm} \end{pmatrix} \cdot \begin{pmatrix} c_0 \\ c_1 \\ \vdots \\ c_m \end{pmatrix} = \begin{pmatrix} b_0 \\ b_1 \\ \vdots \\ b_m \end{pmatrix}, \quad (30)$$

где $\mathbf{A} = \{a_{k\ell}\}$ и $\mathbf{b} = \{b_k\}$ – квадратная матрица и вектор правых частей системы

линейных уравнений, соответственно:

$$a_{k\ell} = \sum_{i=0}^n x_i^k \cdot x_i^\ell, \quad b_k = \sum_{i=0}^n x_i^k \cdot y_i, \quad k, \ell = 0, 1, 2, \dots, m$$

Если среди узлов сетки $\{x_i\}$ нет совпадающих, а также степень полинома меньше чем число узлов аппроксимации $m < n$, то определитель системы не равен нулю $\det \mathbf{A} \neq 0$. Следовательно, эта система имеет единственное решение $\hat{\mathbf{c}} = \{\hat{c}_0, \hat{c}_1, \hat{c}_2, \dots, \hat{c}_m\}$, а полином $p_m(x)$ с такими коэффициентами \hat{c}_i будет обладать минимальным квадратичным отклонением ρ_{\min} .

Зная коэффициенты аппроксимирующего полинома, можно вычислить величину $\|r\|$, например, для сравнения различных аппроксимирующих функций.

Следует отметить, что все коэффициенты $\{c_i\}$ полинома (28) находятся из решения системы уравнений (30), т.е. они связаны между собой. Поэтому если какой-либо коэффициент c_i вследствие его малости ($c_i \approx 0$) отбросить, то необходимо *пересчитывать все оставшиеся коэффициенты* полинома $p_m(x)$.

Кроме того, при изменении даже одного значения исходных данных $\{x_i, y_i\}$ все коэффициенты полинома (28) изменят свои значения, так как они полностью определяются исходными данными. Поэтому при повторении аппроксимации с несколько изменившимися данными (например, вследствие погрешностей измерения, помех, влияния неучтенных факторов и т.п.) получится другая аппроксимирующая функция, отличающаяся коэффициентами.

7.2 Аппроксимирования функций полиномом второй степени $p_2(x)$

Известна таблица данных некоторой функциональной зависимости $y(x)$:

Таблица 12 – Таблично заданная функциональная зависимость $y_i = f(x_i)$

| i | 0 | 1 | 2 | 3 | 4 |
|-------|-------|-------|-------|------|------|
| x_i | −0.76 | −0.48 | −0.09 | 0.22 | 0.55 |
| y_i | 5.15 | 4.39 | 4.10 | 5.71 | 5.30 |

Необходимо аппроксимировать функцию $\{y_i\}$, заданную таблично, ал-

гебраическим полиномом второй степени $p_2(x)$:

$$p_2(x) = c_0 + c_1 \cdot x + c_2 \cdot x^2$$

- 1) Построим меру отклонения полинома $p_2(x)$ от таблично заданной функции $y_i = f(x_i)$ на множестве точек $\{x_0, x_1, x_2, x_3, x_4\}$:

$$\|r\| = \rho(c_0, c_1, c_2) = \sum_{i=0}^4 (c_0 + c_1 \cdot x_i + c_2 \cdot x_i^2 - y_i)^2,$$

где $y_i = f(x_i)$ – значение функции в точке x_i .

- 2) Запишем меру отклонения $\rho(c_0, c_1, c_2)$ в явном виде на основе данных из условия задачи:

$$\begin{aligned} \rho(c_0, c_1, c_2) = & (c_0 + c_1 \cdot (-0.76) + c_2 \cdot (-0.76)^2 - 5.15)^2 + \\ & (c_0 + c_1 \cdot (-0.48) + c_2 \cdot (-0.48)^2 - 4.39)^2 + \\ & (c_0 + c_1 \cdot (-0.09) + c_2 \cdot (-0.09)^2 - 4.10)^2 + \\ & (c_0 + c_1 \cdot (0.22) + c_2 \cdot (0.22)^2 - 5.71)^2 + \\ & (c_0 + c_1 \cdot (0.55) + c_2 \cdot (0.55)^2 - 5.30)^2 \end{aligned}$$

- 3) Определим частную производную от меры отклонений $\rho(c_0, c_1, c_2)$ по аргументу c_0 и приравняем её нулю:

$$\begin{aligned} \frac{\partial \rho}{\partial c_0} = & 2 \cdot (c_0 + c_1 \cdot (-0.76) + c_2 \cdot (-0.76)^2 - 5.15) \cdot 1 + \\ & 2 \cdot (c_0 + c_1 \cdot (-0.48) + c_2 \cdot (-0.48)^2 - 4.39) \cdot 1 + \\ & 2 \cdot (c_0 + c_1 \cdot (-0.09) + c_2 \cdot (-0.09)^2 - 4.10) \cdot 1 + \\ & 2 \cdot (c_0 + c_1 \cdot (0.22) + c_2 \cdot (0.22)^2 - 5.71) \cdot 1 + \\ & 2 \cdot (c_0 + c_1 \cdot (0.55) + c_2 \cdot (0.55)^2 - 5.30) \cdot 1 = 0 \end{aligned}$$

Коэффициенты первой строки матрицы **A** и первый элемент вектора **b**:

$$a_{00} = 1 + 1 + 1 + 1 + 1 = 5$$

$$a_{01} = (-0.76) + (-0.48) + (-0.09) + (0.22) + (0.55) = -0.56$$

$$a_{02} = (-0.76)^2 + (-0.48)^2 + (-0.09)^2 + (0.22)^2 + (0.55)^2 = 1.18$$

$$b_0 = 5.15 + 4.39 + 4.10 + 5.71 + 5.30 = 24.65$$

- 4) Определим частную производную от меры отклонений $\rho(c_0, c_1, c_2)$ по аргументу c_1 и приравняем её нулю:

$$\begin{aligned}\frac{\partial \rho}{\partial c_1} = & 2 \cdot (c_0 + c_1 \cdot (-0.76) + c_2 \cdot (-0.76)^2 - 5.15) \cdot (-0.76) + \\ & 2 \cdot (c_0 + c_1 \cdot (-0.48) + c_2 \cdot (-0.48)^2 - 4.39) \cdot (-0.48) + \\ & 2 \cdot (c_0 + c_1 \cdot (-0.09) + c_2 \cdot (-0.09)^2 - 4.10) \cdot (-0.09) + \\ & 2 \cdot (c_0 + c_1 \cdot (0.22) + c_2 \cdot (0.22)^2 - 5.71) \cdot (0.22) + \\ & 2 \cdot (c_0 + c_1 \cdot (0.55) + c_2 \cdot (0.55)^2 - 5.30) \cdot (0.55) = 0\end{aligned}$$

Коэффициенты второй строки матрицы **A** и второй элемент вектора **b**:

$$\begin{aligned}c_{10} &= (-0.76) + (-0.48) + (-0.09) + (0.22) + (0.55) = -0.56 \\ c_{11} &= (-0.76)^2 + (-0.48)^2 + (-0.09)^2 + (0.22)^2 + (0.55)^2 = 1.18 \\ c_{12} &= (-0.76)^3 + (-0.48)^3 + (-0.09)^3 + (0.22)^3 + (0.55)^3 = -0.38 \\ b_1 &= 5.15 \cdot (-0.76) + 4.39 \cdot (-0.48) + 4.10 \cdot (-0.09) + \\ & 5.71 \cdot (0.22) + 5.30 \cdot (0.55) = -2.24\end{aligned}$$

- 5) Определим частную производную от меры отклонений $\rho(c_0, c_1, c_2)$ по аргументу c_2 и приравняем её нулю:

$$\begin{aligned}\frac{\partial \rho}{\partial c_2} = & 2 \cdot (c_0 + c_1 \cdot (-0.76) + c_2 \cdot (-0.76)^2 - 5.15) \cdot (-0.76)^2 + \\ & + 2 \cdot (c_0 + c_1 \cdot (-0.48) + c_2 \cdot (-0.48)^2 - 4.39) \cdot (-0.48)^2 + \\ & + 2 \cdot (c_0 + c_1 \cdot (-0.09) + c_2 \cdot (-0.09)^2 - 4.10) \cdot (-0.09)^2 + \\ & + 2 \cdot (c_0 + c_1 \cdot (0.22) + c_2 \cdot (0.22)^2 - 5.71) \cdot (0.22)^2 + \\ & + 2 \cdot (c_0 + c_1 \cdot (0.55) + c_2 \cdot (0.55)^2 - 5.30) \cdot (0.55)^2 = 0\end{aligned}$$

Коэффициенты третьей строки матрицы **A** и третий элемент вектора **b**:

$$\begin{aligned}c_{20} &= (-0.76)^2 + (-0.48)^2 + (-0.09)^2 + (0.22)^2 + (0.55)^2 = 1.18 \\ c_{21} &= (-0.76)^3 + (-0.48)^3 + (-0.09)^3 + (0.22)^3 + (0.55)^3 = -0.38 \\ c_{22} &= (-0.76)^4 + (-0.48)^4 + (-0.09)^4 + (0.22)^4 + (0.55)^4 = 0.49 \\ b_2 &= 5.15 \cdot (-0.76)^2 + 4.39 \cdot (-0.48)^2 + 4.10 \cdot (-0.09)^2 + \\ & 5.71 \cdot (0.22)^2 + 5.30 \cdot (0.55)^2 = 5.94\end{aligned}$$

- 6) Таким образом, для определения неизвестных коэффициентов c_0, c_1, c_2 аппроксимирующего полинома $p_2(x)$ необходимо решить систему линейных алгебраических уравнений:

$$\begin{pmatrix} 5 & -0.56 & 1.18 \\ -0.56 & 1.18 & -0.38 \\ 1.18 & -0.38 & 0.49 \end{pmatrix} \cdot \begin{pmatrix} c_0 \\ c_1 \\ c_2 \end{pmatrix} = \begin{pmatrix} 24.65 \\ -2.24 \\ 5.94 \end{pmatrix}$$

- 7) Решение этой системы линейных уравнений можно найти методом Гаусса:

$$\begin{cases} c_0 = 4.66 \\ c_1 = 0.80 \\ c_2 = 1.52 \end{cases}$$

Таким образом, аппроксимирующий полином имеет вид:

$$p_2(x) = 4.66 + 0.80 \cdot x + 1.52 \cdot x^2 \quad (31)$$

- 8) На одном графике представим диаграмму рассеяния (разброса) данных функции заданной таблично $y_i = f(x_i)$ (маркеры) и результаты вычислений аппроксимирующего алгебраического полинома второго порядка $p_2(x)$ (сплошная линия).

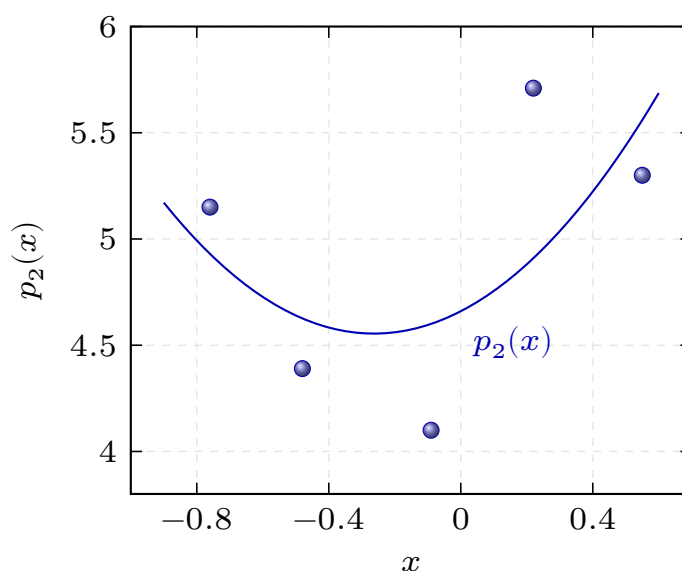


Рисунок 28 – График таблично заданной функции $y_i = f(x_i)$ (маркеры) и аппроксимирующего алгебраического полинома $p_2(x)$ (сплошная линия)

9) Рассчитаем значения аппроксимирующего полинома (31) и ошибку аппроксимации таблично заданной функции в узлах сетки $\{x_i\}$ (таблица 13):

$$\epsilon_i = y_i - p_2(x_i) \quad (32)$$

Таблица 13 – Рассчитанные значения аппроксимирующего полинома $p_2(x)$

| i | 0 | 1 | 2 | 3 | 4 |
|--------------|-------|-------|-------|------|-------|
| x_i | −0.76 | −0.48 | −0.09 | 0.22 | 0.55 |
| y_i | 5.15 | 4.39 | 4.10 | 5.71 | 5.30 |
| $p_2(x_i)$ | 4.93 | 4.63 | 4.60 | 4.91 | 5.56 |
| ϵ_i | 0.22 | −0.24 | −0.50 | 0.80 | −0.26 |

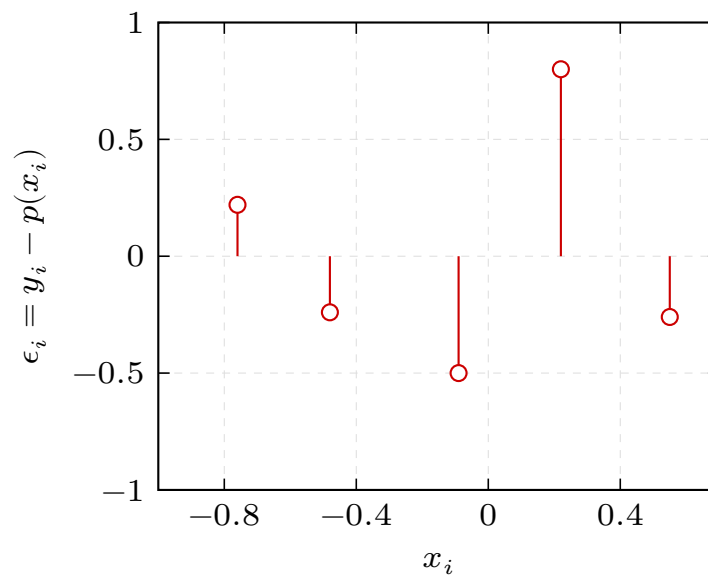


Рисунок 29 – Ошибка аппроксимации полином $p(x)$ функции заданной таблично $y_i = f(x_i)$

10) Рассчитаем среднее квадратичное отклонение δ полинома (31) от значений

функции $\{y_i\}$ в узлах сетки $\{x_i\}$:

$$\delta = \frac{1}{n} \cdot \sum_{i=0}^{n-1} \epsilon_i^2 = \frac{0.05 + 0.06 + 0.25 + 0.64 + 0.07}{5} = 0.213, \quad (33)$$

где $n = 5$ – количество узлов сетки.

8 Задача Коши для систем обыкновенных дифференциальных уравнений

При рассмотрении физических явлений и процессов часто не удается найти непосредственную взаимосвязь между величинами, характеризующими эволюционный, т.е. изменяющийся во времени, процесс. Однако во многих случаях можно установить связь между искомыми характеристиками изучаемого явления (функциями) и скоростями их изменения относительно других переменных, т.е. найти уравнения, в которые входят производные от неизвестных функций. Такие уравнения называют *дифференциальными*.

Задача Коши обычно возникает при анализе процессов, определяемых дифференциальным законом эволюции и начальным состоянием (начальным условием) и для системы обыкновенных дифференциальных уравнений эта задача формулируется в виде системы уравнений:

$$\frac{du}{dt} = f(t, u), \quad u(0) = \dot{u}, \quad (34)$$

где $u(t)$ – неизвестные функции, которые подлежат определению; $f(t, u)$ – известные функции, зависящие от времени и неизвестных функций; \dot{u} – *начальные условия*, т.е. значения неизвестных функций в начальный момент времени ($t = 0$).

Система обыкновенных дифференциальных уравнений первого порядка и начальные условия (34) в развернутом виде могут быть записаны как:

$$\left\{ \begin{array}{l} \frac{du_1}{dt} = f_1(t, u_1, u_2, \dots, u_n), \quad u_1(0) = \dot{u}_1 \\ \frac{du_2}{dt} = f_2(t, u_1, u_2, \dots, u_n), \quad u_2(0) = \dot{u}_2 \\ \dots = \dots \dots \dots \dots \dots \dots = \dots \\ \frac{du_n}{dt} = f_n(t, u_1, u_2, \dots, u_n), \quad u_n(0) = \dot{u}_n \end{array} \right., \quad (35)$$

где n – количество дифференциальных уравнений в системе (34).

Точное решение систем дифференциальных уравнений вида (35), которые описывают многообразие прикладных задач, может быть получено лишь в исключительных случаях. Поэтому возникает необходимость *приближенного решения* таких задач. В настоящее время создано и разработано значительное чис-

ло приближенных методов решения дифференциальных уравнений, каждый из которых имеет свои преимущества и недостатки.

8.1 Метод Эйлера решения задачи Коши

Будем полагать, что решение задачи Коши (35) существует, единственно и обладает необходимыми свойствами гладкости.

Введем временную сетку, т.е. будем рассматривать изменения неизвестных функций только в заданные моменты времени:

$$t = \{t_j\}, \quad j = 0, 1, 2, \dots, m,$$

где j – номер временного интервала; $\Delta t_{j+1} = (t_{j+1} - t_j)$ – шаг сетки, т.е. временной интервал между двумя последовательными моментами времени; m – количество узлов временной сетки.

Основная идея метода Эйлера заключается в предположении, о том что неизвестные функции $u(t)$ изменяются линейно в интервале $[t_j, t_{j+1}]$ между двумя соседними узлами временной сетки и интерполяция неизвестных функций проводится полиномом первого порядка $L_1(t)$:

$$u(t) \approx L_1(t) = \frac{t - t_{j+1}}{t_j - t_{j+1}} \cdot u(t_j) + \frac{t - t_j}{t_{j+1} - t_j} \cdot u(t_{j+1}).$$

Производная от неизвестной функции приближенно аппроксимируется выражением вида:

$$\frac{du}{dt} \approx L'_1(t) = \frac{u(t_{j+1}) - u(t_j)}{t_{j+1} - t_j}, \quad (36)$$

где t_{j+1} и t_j – два последовательных момента времени.

Тогда систему дифференциальных уравнений первого порядка (34) приближенно можно записать в виде:

$$\frac{u(t_{j+1}) - u(t_j)}{\Delta t_{j+1}} \approx f(t_j, u(t_j)). \quad (37)$$

Относительно неизвестных $u(t_{j+1})$ это система линейных алгебраических уравнений и решение системы (37) находится явным образом по рекуррентным

формулам:

$$u(t_{j+1}) = u(t_j) + \Delta t_{j+1} \cdot f(t_j, u(t_j)), \quad u(0) = \dot{u}. \quad (38)$$

Метод Эйлера является простейшим численным методом решения задачи Коши. Блок-схема алгоритма метода Эйлера представлена на рисунке 31. К недостаткам метода можно отнести малую точность и систематическое накопление ошибок.

Для простоты рассмотрим только одно дифференциальное уравнение с единственным начальным условием:

$$y(t_{j+1}) = y(t_j) + \Delta t_{j+1} \cdot f(t_j, y(t_j)), \quad y(0) = \dot{y}$$

На рисунке 30 представлена графическая иллюстрация метода Эйлера численного решения (маркеры) задачи Коши для одного дифференциального уравнения первого порядка (сплошной линией представлено точное решение).

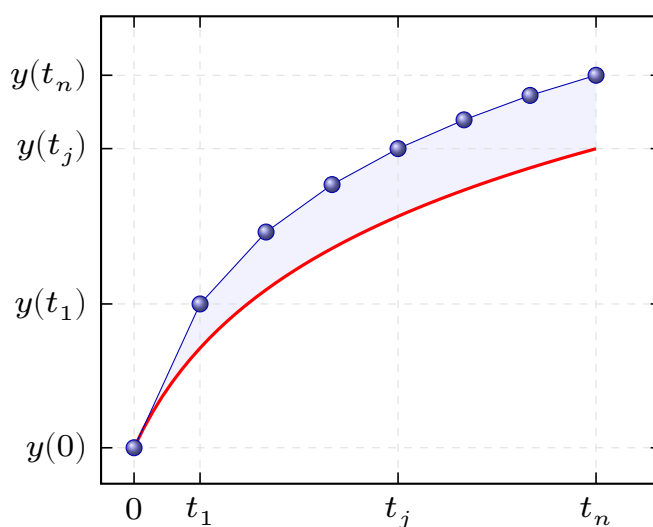


Рисунок 30 – Иллюстрация метода Эйлера:
неизвестная кривая выделена красным цветом, а ее полигональная аппроксимация (ломаная Эйлера) – синим

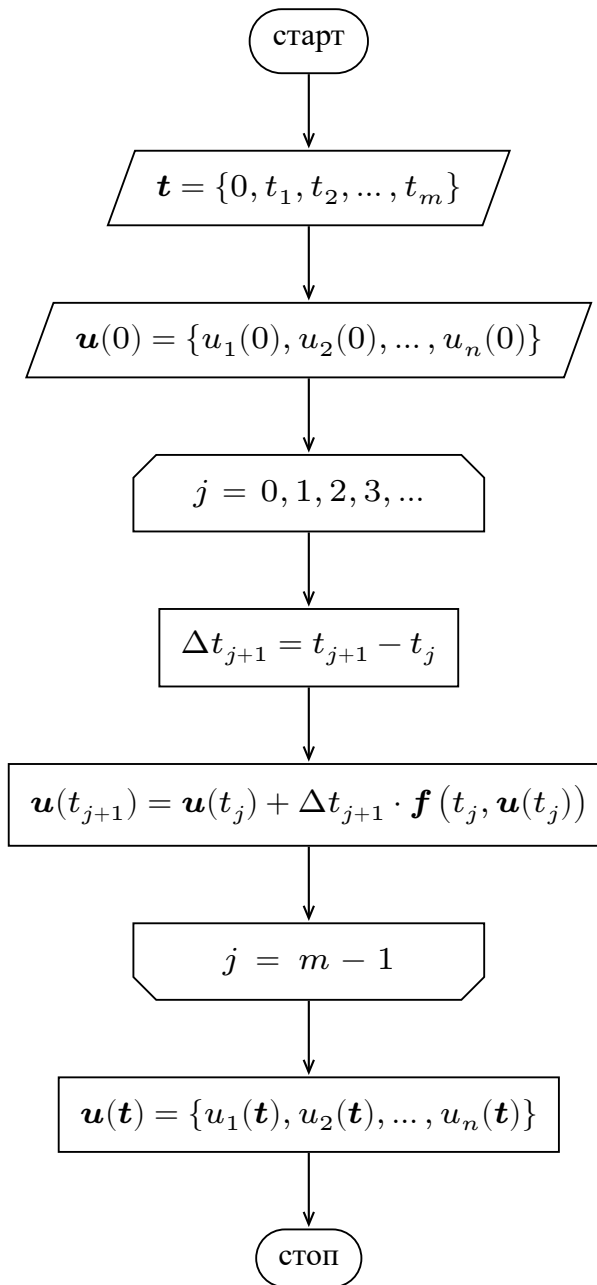


Рисунок 31 – Блок-схема алгоритма численного решения системы дифференциальных уравнений методом Эйлера

8.2 Оценка погрешности решения задачи Коши

Интегрирование системы дифференциальных уравнений (34) по временной переменной t с учетом начальных условий:

$$v(t) = u(0) + \int_0^t f(\xi, v) d\xi. \quad (39)$$

Уравнение (39) является интегральным уравнением для неизвестной функ-

ции $v(t)$, а его решение эквивалентно решению задачи Коши (34), что можно проверить прямой подстановкой (39) в (34).

На временной сетке $\{t_j\}$ интеграл в правой части равенства (39) приближенно вычисляется по формуле трапеций:

$$\int_0^{t_{j+1}} f(\xi, v) d\xi \approx \sum_{k=0}^j \frac{f(t_{k+1}, v(t_{k+1})) + f(t_k, v(t_k))}{2} \cdot (t_{k+1} - t_k). \quad (40)$$

Воспользовавшись (40), выражение для решения интегрального уравнения (39) можно записать в рекуррентной форме:

$$v(t_{j+1}) = v(t_j) + \frac{f(t_{j+1}, v(t_{j+1})) + f(t_j, v(t_j))}{2} \cdot \Delta t_{j+1}. \quad (41)$$

Для определения приближенного значения решения интегрального уравнения (41) могут быть использованы значения неизвестных функций $u(t_j)$, рассчитанные по методу Эйлера (38) на j -ом временном слое:

$$v(t_{j+1}) \approx v(t_j) + \frac{f(t_{j+1}, u(t_{j+1})) + f(t_j, u(t_j))}{2} \cdot \Delta t_{j+1}. \quad (42)$$

В качестве предельной абсолютной погрешности приближенного решения $u(t_j)$ задачи Коши (34) можно принять какую-либо норму:

$$\epsilon(t_j) = v(t_j) - u(t_j) \quad (43)$$

Контроль точности приближенного решения может вестись покомпонентно или по норме. Для различных компонент решения задачи $u(t_j)$ могут использоваться различные допустимые значения погрешности. Контроль точности по норме означает, что контролируется некоторая определенная норма оценки погрешности (рисунок 32):

$$\|\epsilon\|_{\infty} = \max_{i=1..n} |\epsilon_i| \quad \|\epsilon\|_1 = \sum_{i=1}^n |\epsilon_i|, \quad \|\epsilon\|_2 = \sqrt{\sum_{i=1}^n \epsilon_i^2}$$

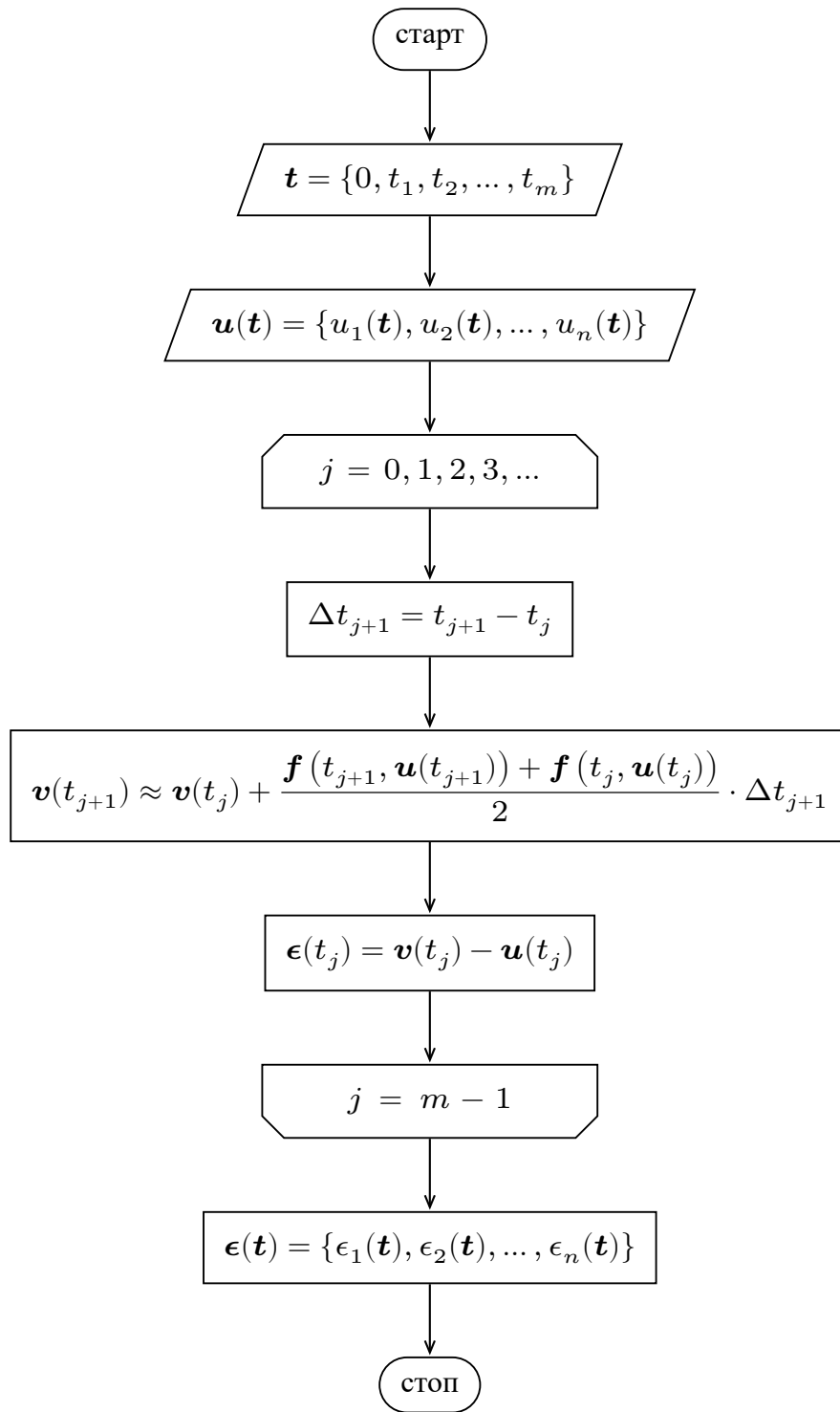


Рисунок 32 – Блок-схема для вычисления погрешности метода Эйлера

8.3 Численное решение задачи Коши методом Эйлера

Применяя метод Эйлера, найдем решение задачи Коши системы дифференциальных уравнений:

$$\begin{cases} \frac{du_1}{dt} = 0.2 \cdot t + u_2, & u_1(0) = 1 \\ \frac{du_2}{dt} = -\frac{u_1}{2}, & u_2(0) = 0 \end{cases}, \quad (44)$$

в пределах отрезка $t \in [0, 10]$ на равномерной сетке с количеством временных интервалов $n = 5$.

Введем обозначения

$$\begin{cases} f_1(t) = 0.2 \cdot t + u_2(t) \\ f_2(t) = -\frac{u_1(t)}{2} \end{cases},$$

где f_1 и f_2 – функции, стоящие в правых частях дифференциальных уравнений системы (44):

Рекуррентные соотношения (38) для решения задачи Коши (44) методом Эйлера:

$$\begin{cases} u_1(t_{j+1}) = u_1(t_j) + \tau \cdot f_1(t_j), & u_1(0) = 1 \\ u_2(t_{j+1}) = u_2(t_j) + \tau \cdot f_2(t_j), & u_2(0) = 0 \end{cases}, \quad (45)$$

где $\tau = (t_{j+1} - t_j)$ – временной шаг метода Эйлера, т.е. интервал между двумя последовательными моментами времени

Решение системы интегральных уравнений (39) определяется рекуррентными соотношениями:

$$\begin{cases} \tilde{u}_1(t_{j+1}) = \tilde{u}_1(t_j) + \frac{f_1(t_j) + f_1(t_{j+1})}{2} \cdot \tau, & \tilde{u}_1(0) = 1 \\ \tilde{u}_2(t_{j+1}) = \tilde{u}_2(t_j) + \frac{f_2(t_j) + f_2(t_{j+1})}{2} \cdot \tau, & \tilde{u}_2(0) = 0 \end{cases}. \quad (46)$$

Определим временной шаг метода Эйлера, зная длину временного отрезка (“время наблюдения”) и количество интервалов:

$$\tau = \frac{T - t_0}{n} = \frac{10 - 0}{5} = 2,$$

где $t_0 = 0$ – начальный момент времени; $T = 10$ – максимальное время (“время наблюдения”).

Введем по переменной t равномерную сетку с шагом $\tau = 2$:

$$t = \{t_j\} = \{0, 2, 4, 6, 8, 10\}$$

Последовательно определяем приближенное решение задачи Коши (44) методом Эйлера, используя рекуррентные соотношения (45).

1) Определим значения неизвестных функций u_1 и u_2 в точке $t_1 = 2$:

$$f_1(0) = 0.2 \cdot 0 + u_2(0) = 0.2 \cdot 0 + 0 = 0$$

$$f_2(0) = -\frac{u_1(0)}{2} = -\frac{1}{2} = -0.5$$

$$\begin{cases} u_1(2) = u_1(0) + 2 \cdot f_1(0) = 1 + 2 \cdot 0 = 1 \\ u_2(2) = u_2(0) + 2 \cdot f_2(0) = 0 + 2 \cdot (-0.5) = -1 \end{cases}.$$

2) Определим значения неизвестных функций u_1 и u_2 в точке $t_2 = 4$:

$$f_1(2) = 0.2 \cdot 2 + u_2(2) = 0.2 \cdot 2 + (-1) = -0.6$$

$$f_2(2) = -\frac{u_1(2)}{2} = -\frac{1}{2} = -0.5$$

$$\begin{cases} u_1(4) = u_1(2) + 2 \cdot f_1(2) = 1 + 2 \cdot (-0.6) = -0.2 \\ u_2(4) = u_2(2) + 2 \cdot f_2(2) = -1 + 2 \cdot (-0.5) = -2 \end{cases}.$$

3) Определим значения неизвестных функций u_1 и u_2 в точке $t_3 = 6$:

$$f_1(4) = 0.2 \cdot 4 + u_2(4) = 0.2 \cdot 4 + (-2) = -1.2$$

$$f_2(4) = -\frac{u_1(4)}{2} = -\frac{-0.2}{2} = 0.1$$

$$\begin{cases} u_1(6) = u_1(4) + 2 \cdot f_1(4) = -0.2 + 2 \cdot (-1.2) = -2.6 \\ u_2(6) = u_2(4) + 2 \cdot f_2(4) = -2 + 2 \cdot (0.1) = -1.8 \end{cases}.$$

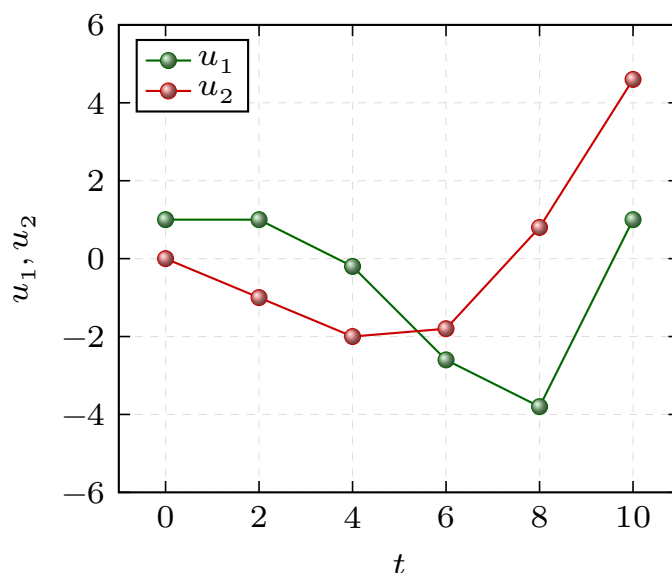


Рисунок 33 – Зависимость неизвестных функций от времени

4) Определим значения неизвестных функций u_1 и u_2 в точке $t_4 = 8$:

$$f_1(6) = 0.2 \cdot 6 + u_2(6) = 0.2 \cdot 6 + (-1.8) = -0.6$$

$$f_2(6) = -\frac{u_1(6)}{2} = -\frac{-2.6}{2} = 1.3$$

$$\begin{cases} u_1(8) = u_1(6) + 2 \cdot f_1(6) = -2.6 + 2 \cdot (-0.6) = -3.8 \\ u_2(8) = u_2(6) + 2 \cdot f_2(6) = -1.8 + 2 \cdot (1.3) = 0.8 \end{cases}.$$

5) Определим значения неизвестных функций u_1 и u_2 в точке $t_5 = 10$:

$$f_1(8) = 0.2 \cdot 8 + u_2(8) = 0.2 \cdot 8 + 0.8 = 2.4$$

$$f_2(8) = -\frac{u_1(8)}{2} = -\frac{-3.8}{2} = 1.9$$

$$\begin{cases} u_1(10) = u_1(8) + 2 \cdot f_1(8) = -3.8 + 2 \cdot (2.4) = 1 \\ u_2(10) = u_2(8) + 2 \cdot f_2(8) = 0.8 + 2 \cdot (1.9) = 4.6 \end{cases}.$$

На рисунке 33 представлено решение задачи Коши системы дифференциальных уравнений (44).

Последовательно определяем приближенное решение интегрального урав-

нения (44), используя рекуррентные соотношения (46).

1) Определим значения неизвестных функций \tilde{u}_1 и \tilde{u}_2 в точке $t_1 = 2$:

$$\begin{cases} \tilde{u}_1(2) = \tilde{u}_1(0) + \tau \cdot \frac{f_1(0) + f_1(2)}{2} = 1 + 2 \cdot \frac{0 + (-0.6)}{2} = 0.4 \\ \tilde{u}_2(2) = \tilde{u}_2(0) + \tau \cdot \frac{f_2(0) + f_2(2)}{2} = 0 + 2 \cdot \frac{-0.5 + (-0.5)}{2} = -1 \end{cases}.$$

2) Определим значения неизвестных функций \tilde{u}_1 и \tilde{u}_2 в точке $t_2 = 4$:

$$\begin{cases} \tilde{u}_1(4) = \tilde{u}_1(2) + \tau \cdot \frac{f_1(2) + f_1(4)}{2} = 0.4 + 2 \cdot \frac{(-0.6) + (-1.2)}{2} = -2.4 \\ \tilde{u}_2(4) = \tilde{u}_2(2) + \tau \cdot \frac{f_2(2) + f_2(4)}{2} = -1 + 2 \cdot \frac{-0.5 + 0.1}{2} = -1.4 \end{cases}.$$

3) Определим значения неизвестных функций \tilde{u}_1 и \tilde{u}_2 в точке $t_3 = 6$:

$$\begin{cases} \tilde{u}_1(6) = \tilde{u}_1(4) + \tau \cdot \frac{f_1(4) + f_1(6)}{2} = -2.4 + 2 \cdot \frac{-1.2 + (-0.6)}{2} = -4.2 \\ \tilde{u}_2(6) = \tilde{u}_2(4) + \tau \cdot \frac{f_2(4) + f_2(6)}{2} = -1.4 + 2 \cdot \frac{0.1 + 1.3}{2} = 0 \end{cases}.$$

4) Определим значения неизвестных функций \tilde{u}_1 и \tilde{u}_2 в точке $t_4 = 8$:

$$\begin{cases} \tilde{u}_1(8) = \tilde{u}_1(6) + \tau \cdot \frac{f_1(6) + f_1(8)}{2} = -4.2 + 2 \cdot \frac{-0.6 + 2.4}{2} = -2.4 \\ \tilde{u}_2(8) = \tilde{u}_2(6) + \tau \cdot \frac{f_2(6) + f_2(8)}{2} = 0 + 2 \cdot \frac{1.3 + 1.9}{2} = 3.2 \end{cases}.$$

5) Определим значения неизвестных функций \tilde{u}_1 и \tilde{u}_2 в точке $t_5 = 10$:

$$f_1(10) = 0.2 \cdot 10 + u_2(10) = 0.2 \cdot 8 + 4.6 = 6.2$$

$$f_2(10) = -\frac{u_1(10)}{2} = -\frac{1}{2} = -0.5$$

$$\begin{cases} \tilde{u}_1(10) = \tilde{u}_1(8) + \tau \cdot \frac{f_1(8) + f_1(10)}{2} = -2.4 + 2 \cdot \frac{2.4 + 6.2}{2} = 6.2 \\ \tilde{u}_2(10) = \tilde{u}_2(8) + \tau \cdot \frac{f_2(8) + f_2(10)}{2} = 3.2 + 2 \cdot \frac{1.9 + (-0.5)}{2} = 4.6 \end{cases}.$$

На рисунке 34 представлены решения задачи Коши (34) и интегрального

уравнения (39), рассчитанные в различные моменты времени.

В таблице 14 и на рисунке 35 представлены значения предельной абсолютной погрешности приближенного решения задачи Коши для различных моментов времени.

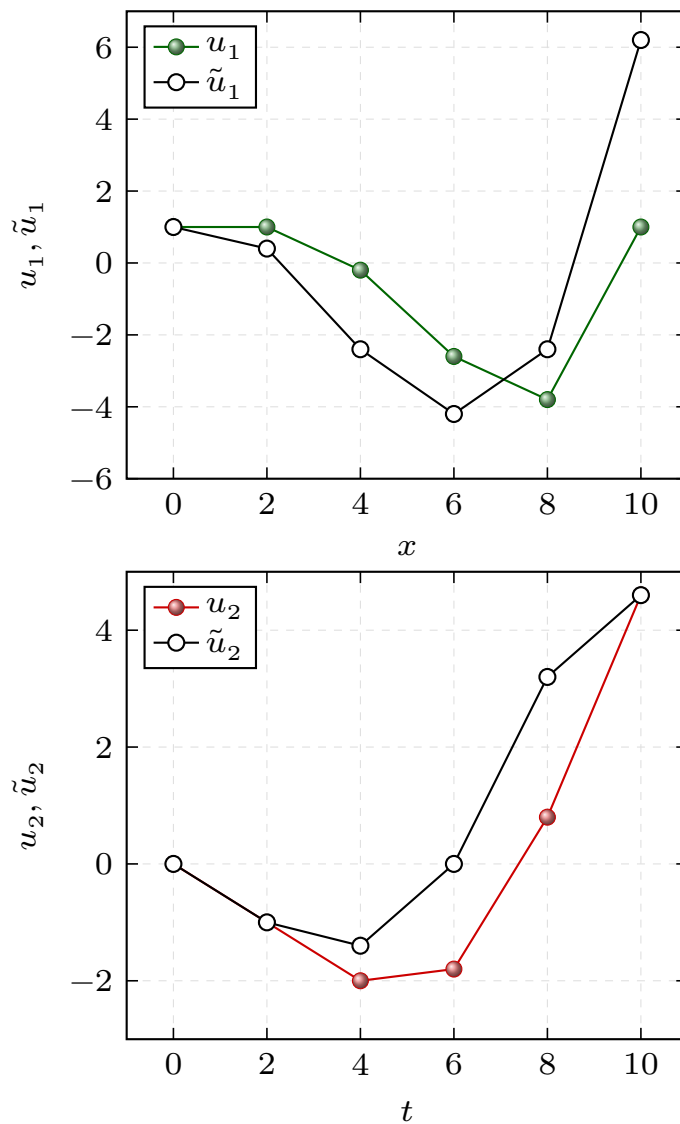


Рисунок 34 – Приближенное решение задачи Коши и соответствующего интегрального уравнения

Таблица 14 – Предельная абсолютная погрешность приближенного решения задачи Коши (44)

| | | | | | | |
|---|---|-----|------|------|------|-----|
| i | 0 | 1 | 2 | 3 | 4 | 5 |
| t_i | 0 | 2 | 4 | 6 | 8 | 10 |
| Задача Коши | | | | | | |
| $u_1(t_i)$ | 1 | 1 | -0.2 | -2.6 | -3.8 | 1 |
| $u_2(t_i)$ | 0 | -1 | -2 | -1.8 | 0.8 | 4.6 |
| Интегральное уравнение | | | | | | |
| $\tilde{u}_1(t_i)$ | 1 | 0.4 | -2.4 | -4.2 | -2.4 | 6.2 |
| $\tilde{u}_2(t_i)$ | 0 | -1 | -1.4 | 0 | 3.2 | 4.6 |
| Абсолютная погрешность $\epsilon = \tilde{u} - u $ | | | | | | |
| ϵ_1 | 0 | 0.6 | 2.2 | 1.6 | 1.4 | 5.2 |
| ϵ_2 | 0 | 0 | 0.6 | 1.8 | 2.4 | 0 |

Из рисунка 35 видно, что максимальная предельная абсолютная погрешность для $u_1(t)$ составляет $\epsilon_1 = 5.2$, а для функции $u_2(t) - \epsilon_2 = 2.4$.

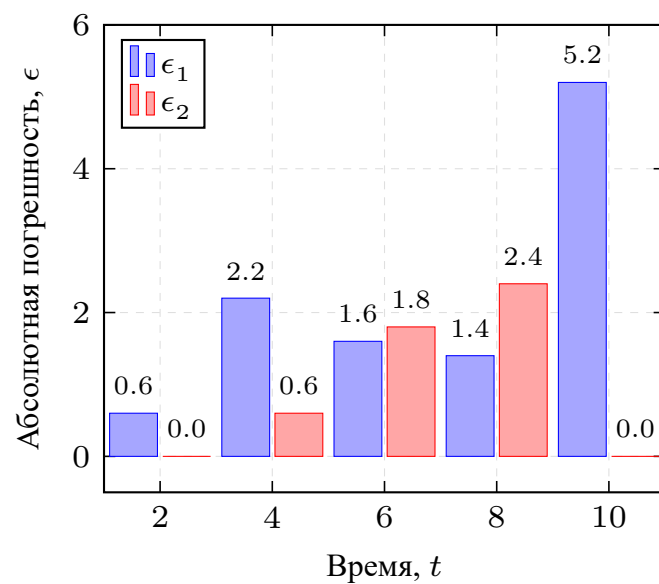


Рисунок 35 – Предельная абсолютная погрешность приближенного решения задачи Коши (44)

9 Краевые задачи для обыкновенных дифференциальных уравнений

Исследование большого круга естественно-научных и инженерных проблем приводит к математическим задачам, относящимся к решению дифференциальных уравнений и граничных проблем для них, интегральных и других функциональных уравнений.

В практике чаще всего встречаются задачи, для которых точное решение не может быть найдено или оно имеет невысокую эффективность. Поэтому приближенные методы решения задач граничных задач, например, задач математической физики, в особенности метод сеток получили широкое распространение.

Основные *достоинства приближенных методов* заключаются в том, что они являются *универсальными и эффективными*, так как позволяют находить приближенное решение для широкого класса задач новейших областей науки, техники и технологий. Как правило, такие задачи формулируются в виде основного дифференциального уравнения (системы дифференциальных уравнений) и дополнительных (граничных) условий, которые обеспечивают существование и единственность решения.

Под краевой (граничной) задачей понимают задачу о нахождении решения дифференциального уравнения (системы дифференциальных уравнений), удовлетворяющего краевым условиям на границе рассматриваемой области.

В каждой краевой задаче искомое решение должно удовлетворять дифференциальному в рассматриваемой области Ω , а также некоторому дополнительному условию на границе $\partial\Omega$ или в нескольких различных точках этой области. Например для одномерных задач, когда искомая функция зависит только от одной пространственной переменной $u(x)$, краевые задачи можно сформулировать в виде:

$$\frac{du(x)}{dx} = f(x, u), \quad \varphi(\xi, u(\xi)) = 0, \quad \xi \in \mathcal{A} \quad (47)$$

где $u(x) = (u_1, u_2, \dots, u_n)^T$ – искомая неизвестная вектор-функция; x – независимое переменное; $f(x, u) = (f_1, f_2, \dots, f_n)^T$ – вектор-функция многих переменных, определенная в некоторой замкнутой области изменения своих аргументов; $\varphi(\xi, u) = (\varphi_1, \varphi_2, \dots, \varphi_n)^T$ – вектор-функция многих переменных,

определяющая дополнительные условия для неизвестной функции в различных точках пространства; $\mathcal{A} = \partial\Omega \cup \Omega$ – множество точек на границе $\partial\Omega$ и/или внутренних точках замкнутой области Ω .

Однако, существуют задачи с более сложными по форме дополнительными условиями, например, условиями нормировки стандартными для задач квантовой механики:

$$\int_{\mathcal{A}} \|\mathbf{u}(\xi)\|^2 d\xi = 1,$$

где $\|\mathbf{u}(\xi)\|$ – норма, заданная в некотором функциональном пространстве; \mathcal{A} – область интегрирования, которая в общем случае бесконечна $\mathcal{A} \rightarrow \infty$.

Формулировка корректно поставленных краевых задач возможна для систем дифференциальных уравнений, имеющих порядок не ниже второго.

Найти точное решение краевой задачи (47) в элементарных функциях возможно в крайне узких классах краевых задач, т.к. необходимо найти не только общее решение системы дифференциальных уравнений, но и явно определить из краевых условий значения входящих в общее решение постоянных коэффициентов. Поэтому для численного решения краевых задач применяется метод стрельбы и разностный метод.

Метод стрельбы основан на формулировке задачи Коши, которая эквивалентна краевой задаче (47). Необходимо отметить, что в отличие от задачи Коши, для которой при выполнении определенных условий гарантируется существование и единственность решения, *краевая задача* для того же дифференциального уравнения *может не иметь решения* или иметь несколько решений (в том числе и бесконечное множество решений).

В разностном методе задача приближенно заменяется решением алгебраической системы уравнений, в которой в качестве неизвестных выступают значения решения в узлах пространственной сетки.

В случае нелинейных задач оба метода являются итерационными, причем построение быстро сходящихся итерационных процессов само является достаточно сложной задачей.

9.1 Линейные краевые условия

Для систем дифференциальных уравнений специального вида

$$\begin{cases} w'(x) = f_1(x, u, w) \\ u'(x) = f_2(x) \cdot w(x) \end{cases} \quad (48)$$

в зависимости от вида условий на границах отрезка $0 \leq x \leq \ell$ (граница области поиска решений $\partial\Omega$) различают три основных типа граничных задач: задачу Дирихле, Неймана и смешанную краевую задачу (таблица 15).

Таблица 15 – Основные виды краевых условий для системы обыкновенных дифференциальных уравнений (48)

| № | Краевое условие | Граница $x = 0$ | Граница $x = \ell$ |
|---|-----------------|---|---|
| 1 | Дирихле | $u(0) = u_0$ | $u(\ell) = u_1$ |
| 2 | Неймана | $w(0) = w_0$ | $w(\ell) = w_1$ |
| 3 | Смешанная | $a_0 \cdot w(0) + b_0 \cdot u(0) = c_0$ | $a_1 \cdot w(\ell) + b_1 \cdot u(\ell) = c_1$ |

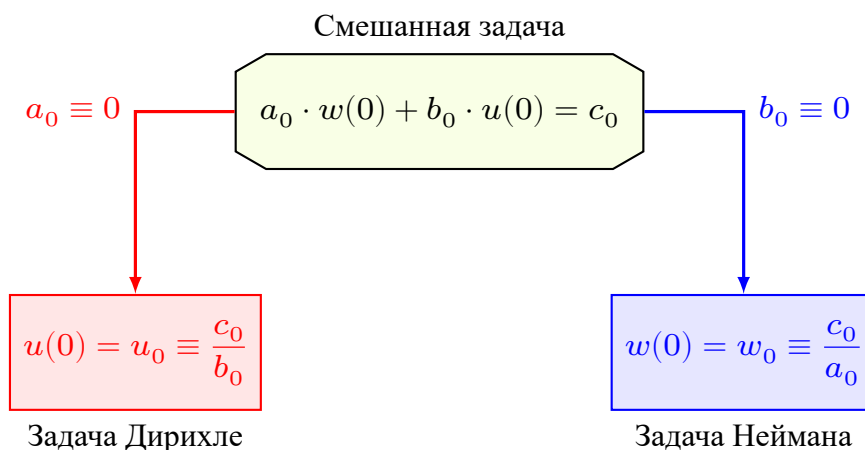


Рисунок 36 – Виды краевых задач для систем обыкновенных дифференциальных уравнений вида (48)

С формальной точки зрения, задача Дирихле и Неймана являются частны-

ми случаями смешанной краевой задачи (рисунок 36): например, если $a_0 \equiv 0$, то смешанная краевая задача трансформируется в задачу Дирихле:

$$0 \cdot w(0) + b_0 \cdot u(0) = c_0, \quad \Rightarrow \quad u(0) = \frac{c_0}{b_0} = u_0,$$

а если полагать $b_0 \equiv 0$, то формулируется задача Неймана:

$$a_0 \cdot w(0) + 0 \cdot u(0) = c_0, \quad \Rightarrow \quad w(0) = \frac{c_0}{a_0} = w_0.$$

Поэтому для общности, можно рассматривать только смешанную краевую задачу стандартного вида:

$$\begin{cases} w'(x) = f_1(x, u, w) \\ u'(x) = f_2(x) \cdot w(x) \\ a_0 \cdot w(0) + b_0 \cdot u(0) = c_0 \\ a_1 \cdot w(\ell) + b_1 \cdot u(\ell) = c_1 \end{cases}, \quad (49)$$

где $f_1(x, u, w)$ и $f_2(x)$ – известные функции, в том числе может быть разрывные; $a_{0,1}$, $b_{0,1}$ и $c_{0,1}$ – известные постоянные коэффициенты, определяемые физическим смыслом решаемой задачи.

9.2 Построение разностных схем методом баланса

Различные физические процессы (теплопроводности или диффузии, колебаний, газодинамики и т. д.) характеризуются некоторыми интегральными законами сохранения (тепла, массы, количества движения, энергии и т. д.). При выводе дифференциальных уравнений различных математических моделей объектов или явлений отталкиваются от некоторого интегрального соотношения (уравнения баланса), выражающего закон сохранения для малого объема. Дифференциальное уравнение получается из уравнения баланса при стягивании рассматриваемого объема к нулю в предположении существования непрерывных производных, входящих в уравнение.

Метод конечных разностей физически означает переход от непрерывной среды к некоторой ее дискретной модели. При таком переходе естественно требовать, чтобы основные свойства физического процесса сохранялись, т.е. выполнялись законы сохранения различных физических субстанций (количества тепла, массы, заряда, импульса и т.д.).

Численные методы решения краевых задач основаны на замене дифференциальной задачи (49) системой линейных алгебраических уравнений, т.е. разностной схемой.

Разностная схема – совокупность разностных алгебраических уравнений, аппроксимирующих систему дифференциальных уравнений и дополнительные (граничные) условия исходной задачи. Разностные схемы, выражающие на сетке законы сохранения физических субстанций, называют консервативными (или дивергентными). Законы сохранения для всей сеточной области (“интегральные законы сохранения”) для консервативных схем должны быть алгебраическим следствием разностных уравнений.

Для получения консервативных разностных схем естественно исходить из уравнений баланса, записанных для элементарных объемов (ячеек) сеточной области. Входящие в эти уравнения балансы интегралы и производные следует заменить приближенными разностными выражениями. В результате получаем однородную разностную схему. Такой метод получения консервативных однородных разностных схем называется интегро-интерполяционным методом (методом баланса).

9.2.1 Физическая интерпретация краевых задач

Уравнение (49) можно интерпретировать как уравнение установившегося распределения температуры $u(x)$ в стержне длины ℓ . В такой постановке краевую задачу можно записать в виде:

$$\begin{cases} w'(x) + r(x) \cdot u(x) = q(x) \\ w(0) + p_0 \cdot u(0) = s_0 \\ -w(\ell) + p_\ell \cdot u(\ell) = s_\ell \end{cases}, \quad (50)$$

где $w(x)$ – тепловой поток, величина которого в соответствии с [законом теплопроводности Фурье](#) пропорциональна градиенту температуры:

$$w(x) = -k(x) \cdot u'(x); \quad (51)$$

$k(x)$ – коэффициент температуропроводности материала стержня; $r(x) \cdot u(x)$ – мощность распределенных стоков ($r > 0$) или источников ($r < 0$) тепла вдоль боковой поверхности стержня; p_ℓ – коэффициент конвекционного теплооб-

мена боковой поверхности стержня с окружающей средой; $q(x)$ – мощность внутренних распределенных источников ($q > 0$) или стоков ($q < 0$) тепла в стержне.

Граничные условия (50) определяют процесс теплообмена концов стержня $x = \{0, \ell\}$ с окружающей средой по закону **конвекционного теплообмена Ньютона** – “Тепловой поток тепла \dot{w} через поверхность материального тела в окружающую среду прямо пропорционален разнице температур между поверхностью тела и окружающей средой”:

$$\dot{w} = r \cdot (u - \dot{u}),$$

где r – коэффициент конвективного теплообмена; u – температура поверхности тела; \dot{u} – температура окружающей среды.

Процессы выделения или поглощения тепла в твердом теле могут быть связаны с рядом явлений:

- 1) выделение тепла ($q(x) > 0$) при пропускании электрического тока через стержень (закон Джоуля-Ленца), вследствие протекания в объеме стержня каких-либо экзотермических химических реакций;
- 2) поглощение тепла ($q(x) < 0$) за счет термоэлектрических явлений (например, эффект Пельтье), а также вследствие протекания эндотермических химических реакций в объеме материала стержня.

Необходимо отметить, что в зависимости от физического смысла постановки краевых задач коэффициенты дифференциального уравнения задачи (50) могут быть и разрывными функциями. Можно выделить несколько практически важных случаев:

- 1) *однородный стержень* с локальной теплоизоляцией, т.е. конвекционный теплообмен участка $x \in [a, b]$ боковой поверхности стержня с окружающей средой затруднен или отсутствует (рисунок 37а):

$$r(x) = \begin{cases} r_1, & 0 \leq x < a \\ r_2, & a \leq x \leq b, \\ r_1, & b < x \leq \ell \end{cases}$$

где $[a, b]$ – теплоизолированный участок однородного стержня;

- 2) *однородный стержень* с локальным нагревом, т.е. на участке стержня $[a, b]$

происходит нагрев стержня за счет внешних источников тепловой энергии (рисунок 37b), например, излучения:

$$q(x) = \begin{cases} 0, & 0 \leq x < a \\ q_1, & a \leq x \leq b, \\ 0, & b < x \leq \ell \end{cases}$$

где $[a, b]$ – участок локального нагрева стержня внешними источниками тепловой энергии;



(a) теплоизолированный участок



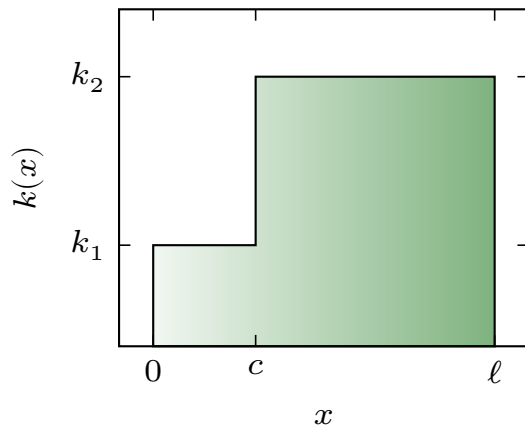
(b) локальный нагрев

Рисунок 37 – Однородный образец

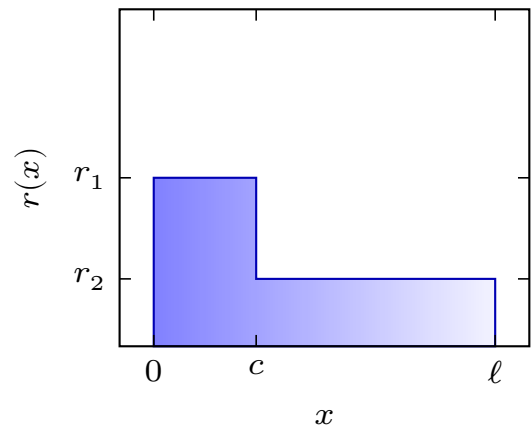
3) *неоднородный стержень*, состоящий из нескольких разнородных материалов, которые отличаются коэффициентами температуропроводности $k_1 \neq k_2$ и конвекционного теплообмена $r_1 \neq r_2$ (рисунок):

$$k(x) = \begin{cases} k_1, & 0 \leq x \leq c \\ k_2, & c < x \leq \ell \end{cases}, \quad r(x) = \begin{cases} r_1, & 0 \leq x \leq c \\ r_2, & c < x \leq \ell \end{cases},$$

где c – точка контакта двух различных материалов; $[0, c]$ и $(c, \ell]$ – участки стержня, состоящие из разных материалов с различными коэффициентами температуропроводности $k_1 \neq k_2$ и конвекционного теплообмена $r_1 \neq r_2$;



(a) теплопроводность



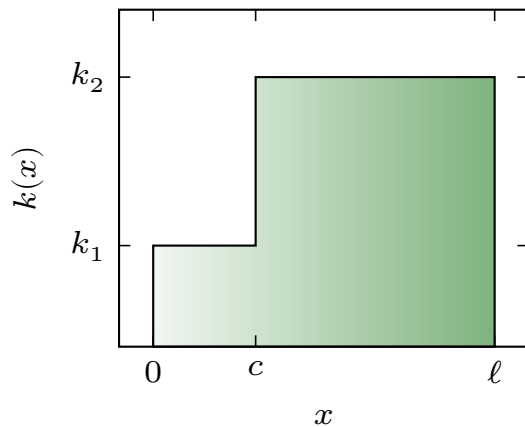
(b) конвекционный теплообмен

Рисунок 38 – Неоднородный образец

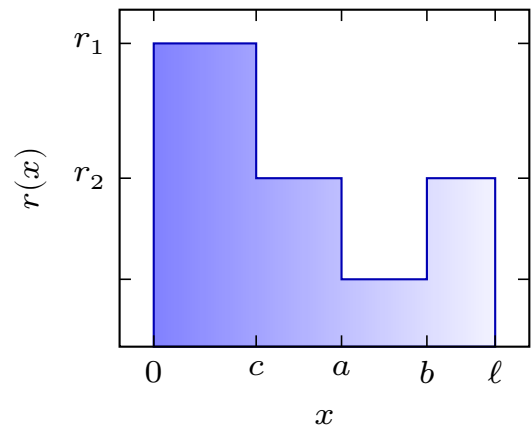
- 4) *неоднородный стержень*, с локальной теплоизоляцией, т.е. конвекционный теплообмен на участке боковой поверхности стержня $[a, b]$ с окружающей средой затруднен или отсутствует (рисунок):

$$k(x) = \begin{cases} k_1, & 0 \leq x \leq c \\ k_2, & c < x \leq \ell \end{cases}, \quad r(x) = \begin{cases} r_1, & 0 \leq x \leq a \\ r_2, & a < x \leq b \\ r_3, & b < x \leq \ell \end{cases},$$

где $[a, b]$ – теплоизолированный участок неоднородного стержня ($r \equiv 0$).



(a) теплопроводность



(b) конвекционный теплообмен

Рисунок 39 – Неоднородный образец с локальной теплоизоляцией

9.2.2 Расчётная сетка

При численном решении той или иной математической задачи невозможно получить решение для всех значений аргумента, изменяющегося внутри некоторой области евклидова пространства. Поэтому в этой области выбирается некоторое конечное множество точек и отыскивается приближенное решение только в этих точках. Такое множество точек называется сеткой, а отдельные точки – узлами сетки. Необходимо отметить, что свойства разностного решения задачи и, в частности, его близость к точному решению зависят от выбора сетки.

Процедуру построения расчетной сетки можно рассматривать как построение взаимно-однозначного отображения области определения функции (физической области) на некоторую расчетную область более простой формы.

Для перехода от дифференциальной краевой задачи (50) к системе алгебраических уравнений на отрезке $[0, \ell]$ введем произвольную неравномерную сетку – конечное упорядоченное множество точек $\{x_i\}$, принадлежащих этому отрезку (рисунок 40):

$$0 = x_0 < x_1 < x_2 < \dots < x_i < \dots < x_{n-1} < x_n = \ell,$$

где $x_i \in [0, \ell]$ – узлы сетки; $i = 0, 1, 2, \dots, n$ – порядковый номер узла сетки; n – количество узлов сетки.

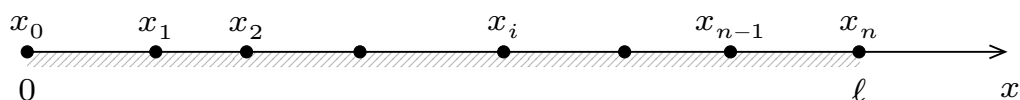


Рисунок 40 – Схематическое изображение расположения узлов x_i расчетной сетки $\{x_0, x_1, \dots, x_n\}$

Если известны точки разрывов коэффициентов $f(x) = \{k(x), r(x), q(x)\}$ системы дифференциальных уравнений (50), то всегда можно выбрать неравномерную сетку так, чтобы эти точки разрывов совпадали с узлами расчетной сетки (рисунок 41).

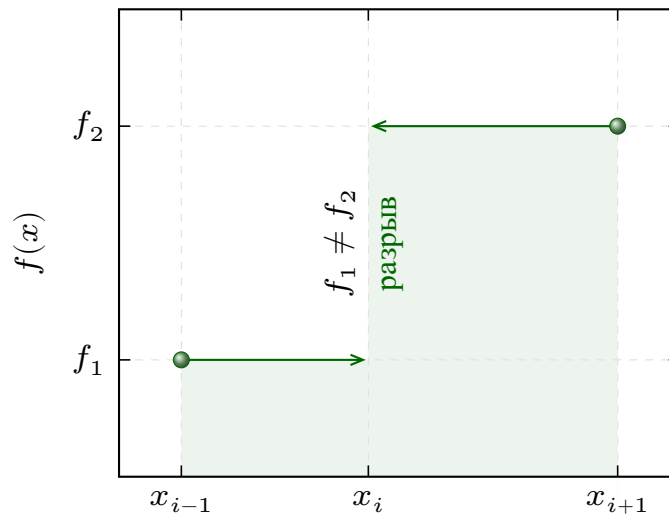


Рисунок 41 – График функции $f(x)$, которая имеет разрыв в точке $x = x_i$

Кроме того, всегда можно построить локально равномерную сетку по алгоритму:

- 1) Строится равномерная сетка $\{\hat{x}_i\}$ во всей расчётной области $[0, \ell]$:

$$\hat{x}_i = \frac{i}{n} \cdot \ell, \quad i = 0, 1, \dots, n,$$

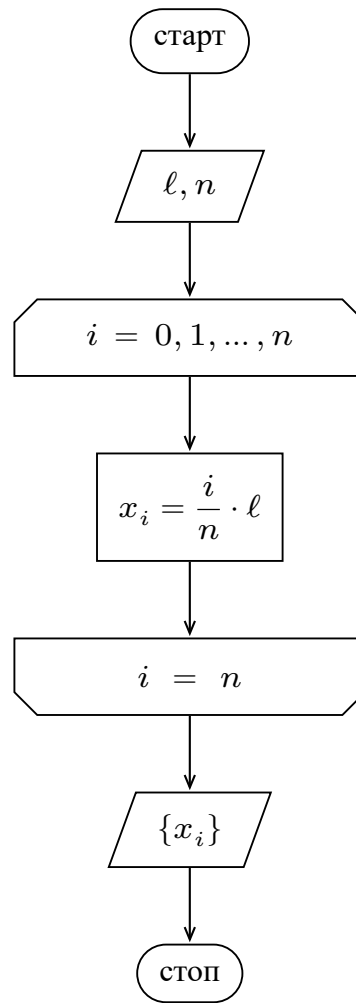


Рисунок 42 – Блок-схема алгоритма построения равномерной расчетной сетки $\{x_i\} = (0, x_1, x_2, \dots, x_{n-1}, \ell)$ с заданным количеством интервалов n

- 2) Упорядочиваются все точки разрыва ξ_j ($j = 0, 1, 2, \dots, m$) коэффициентов краевой задачи, включая левую ($\xi_0 = 0$) и правую ($\xi_{m+1} = \ell$) границы расчётной области (рисунок 43):

$$0 = \xi_0 < \xi_1 < \xi_2 < \dots < \xi_m < \xi_{m+1} = \ell,$$

где m – количество внутренних точек разрыва коэффициентов краевой задачи.

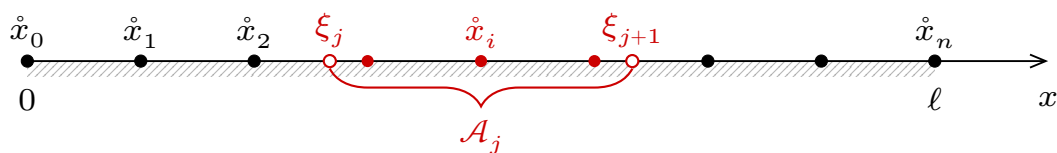


Рисунок 43 – Схематическое расположение узлов равномерной сетки $\{\dot{x}_i\}$

- 3) Определяются области непрерывности $\mathcal{A}_j = [\xi_j, \xi_{j+1})$ всех коэффициентов краевой задачи в расчётной области:

$$\mathcal{A}_0 \cup \mathcal{A}_1 \cup \dots \cup \mathcal{A}_m = [0, \ell].$$

- 4) Находится количество узлов m_j равномерной сетки $\{\hat{x}_i\}$, которые принадлежат каждой области непрерывности:

$$m_j = m_j + \begin{cases} 1 & \text{if } \hat{x}_i \in \mathcal{A}_j \\ 0 & \text{otherwise} \end{cases}.$$

- 5) Строятся равномерные сетки в пределах каждой области непрерывности \mathcal{A}_j :

$$\{\hat{x}_i\}_j = \xi_j + \frac{i}{m_j} \cdot (\xi_{j+1} - \xi_j), \quad i = 0, 1, \dots, m_j - 1,$$

- 6) Формируется локально равномерная сетка с помощью объединения всех равномерных сеток и правой границы расчётной области:

$$\{x_i\} = \{\hat{x}_i\}_0 \cup \{\hat{x}_i\}_1 \cup \dots \cup \{\hat{x}_i\}_m \cup \ell$$

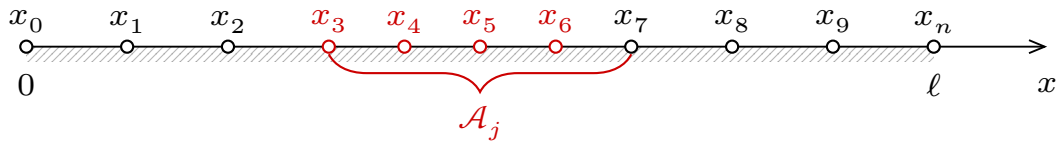


Рисунок 44 – Расположение узлов *локально равномерной* сетки $\{x_i\}$

Выделим на отрезке $[0, \ell]$ подмножество промежуточных узлов $\{x_{i\pm 1/2}\}$ (рисунок 45):

$$x_{i\pm 1/2} = \frac{x_i + x_{i\pm 1}}{2}, \quad i = 1, 2, \dots, n - 1,$$

где $x_{i\pm 1/2}$ – середины отрезков $[x_{i-1}, x_i]$ и $[x_i, x_{i+1}]$, соответственно.

Значения неизвестной функции $u(x)$ (температуры) рассмотрим в узлах сетки $\{x_i\}$, а потоковую величину $w(x)$ (тепловой поток) – в промежуточных узлах $\{x_{i\pm 1/2}\}$.

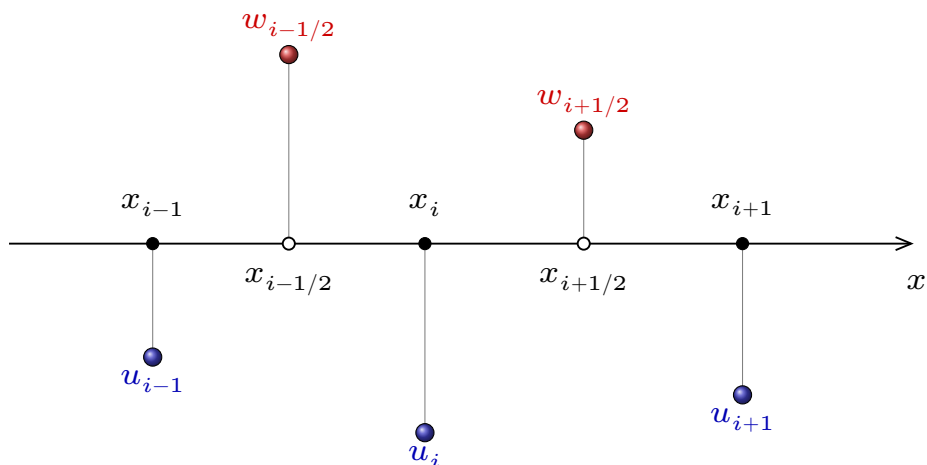


Рисунок 45 – Схематическое изображение расположения узлов основной $\{x_{i\mp 1}\}$ и промежуточной $\{x_{i\mp 1/2}\}$ расчетной сеток

9.2.3 Разностная схема для дифференциального уравнения

Для построения разностной схемы для краевой задачи воспользуемся интегро-интерполяционным методом (или *методом баланса*) построения разностных схем. Проинтегрируем *дифференциальное уравнение* (50) в пределах контрольной области $[x_{i-1/2}, x_{i+1/2}]$:

$$\int_{x_{i-1/2}}^{x_{i+1/2}} w'(x) dx + \int_{x_{i-1/2}}^{x_{i+1/2}} r(x) \cdot u(x) dx = \int_{x_{i-1/2}}^{x_{i+1/2}} q(x) dx \quad (52)$$

Тогда уравнение теплового баланса для контрольной области (52) запишется в виде:

$$w(x_{i+1/2}) - w(x_{i-1/2}) + \int_{x_{i-1/2}}^{x_{i+1/2}} r(x) \cdot u(x) dx = \int_{x_{i-1/2}}^{x_{i+1/2}} q(x) dx \quad (53)$$

На рисунке 46 представлена графическая иллюстрация процессов переноса тепла, теплообмена и тепловыделения (телопоглощения) для контрольной области $[x_{i-1/2}, x_{i+1/2}]$.

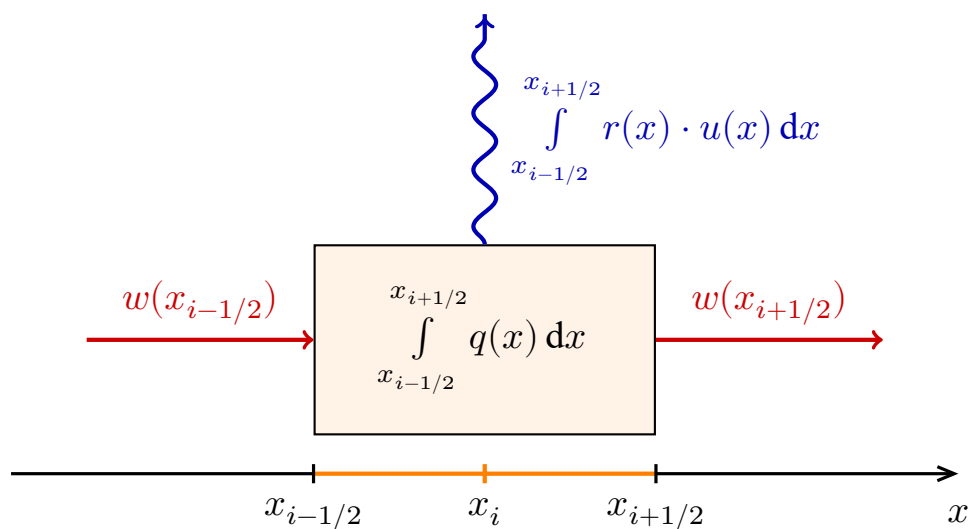


Рисунок 46 – Иллюстрация баланса тепла (53) для контрольной области $[x_{i-1/2}, x_{i+1/2}]$

Рассмотрим физический смысл каждого члена уравнения теплового баланса (53):

- 1) первое слагаемое $w(x_{i-1/2})$ определяет количество тепла, “втекающего” через сечение $x = x_{i-1/2}$;
- 2) второе слагаемое $w(x_{i+1/2})$ это количество “вытекающего” тепла через сечение $x = x_{i+1/2}$;
- 3) третье слагаемое в левой части представляет собой количество тепла, отдаваемое стержнем внешней среде за счет конвекционного теплообмена на его боковой поверхности;
- 4) правая часть (53) соответствует количеству тепла, выделяющегося на отрезке $[x_{i-1/2}, x_{i+1/2}]$ за счет распределенных источников тепла с плотностью $q(x)$.

Для определения теплового потока в промежуточных узлах сетки $w(x_{i\pm 1/2})$, воспользуемся закон теплопроводности Фурье (51), из которого следует:

$$u'(x) = -w(x) \cdot k^{-1}(x) \quad \Rightarrow \quad \int_{x_i}^{x_{i+1}} u'(x) dx = - \int_{x_i}^{x_{i+1}} w(x) \cdot k^{-1}(x) dx.$$

После интегрирования левой части:

$$u(x_{i+1}) - u(x_i) = - \int_{x_i}^{x_{i+1}} w(x) \cdot k^{-1}(x) dx. \quad (54)$$

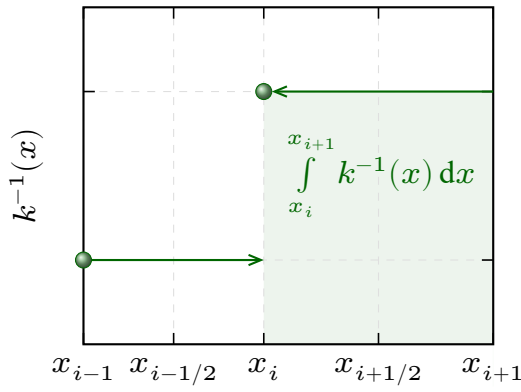
При построении разностной схемы, в выражениях (53) и (54) необходимо вычислять определенные интегралы. Однако, во многих практически важных приложениях первообразные подынтегральных функций не могут быть выражены в элементарных функциях. Кроме того, коэффициенты краевой задачи могут быть известны только в узлах расчетной сетки $\{k(x_i), r(x_i), q(x_i)\}$. В этом случае для вычислений определенных интегралов необходимо пользоваться *численным методом интегрирования*.

Для этого воспользуемся простейшими интерполяциями коэффициентов $f(x) = \{k(x), r(x), q(x)\}$ (рисунок 41), неизвестной функции $u(x)$ (температуры) и ее потока $w(x)$ (тепловой поток) в окрестности узлов $\{x_i\}$ в виде:

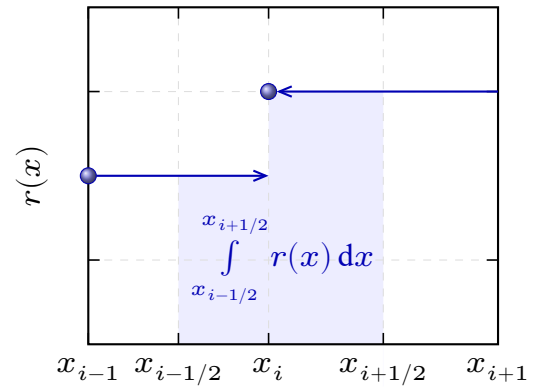
$$\begin{cases} w(x) = w(x_{i+1/2}), & x_i \leq x \leq x_{i+1} \\ f(x) = f(x_i), & x_i \leq x < x_{i+1} \\ u(x) = u(x_i), & x_{i-1/2} \leq x \leq x_{i+1/2} \end{cases}. \quad (55)$$

Тогда левосторонняя формула прямоугольников совпадает с точным значением интегралов (рисунок 47):

$$\int_{x_i}^{x_{i+1}} y(x) dx \approx y(x_i) \cdot (x_{i+1} - x_i). \quad (56)$$



(a) теплопроводность



(b) конвекционный теплообмен

Рисунок 47 – Приближенное вычисление значения определенных интегралов в уравнении баланса тепла (53) и законе Фурье (54)

Приближенное значение интеграла в (54) выразим воспользовавшись кусочной аппроксимацией (55) теплового потока $w(x) \approx w(x_{i+1/2})$ на интервале (x_i, x_{i+1}) :

$$u(x_{i+1}) - u(x_i) \approx -w(x_{i+1/2}) \cdot \int_{x_i}^{x_{i+1}} k^{-1}(x) dx \quad (57)$$

Применяя *левостороннюю формулу прямоугольников* для вычисления интеграла (56), найдем поток тепла в промежуточных узлах сетки $w(x_{i+1/2})$:

$$w_{i+1/2} = -\lambda_i \cdot (u_{i+1} - u_i) \quad (58)$$

где $w_{i+1/2}$ – величина теплового потока в узле вспомогательной сетки $x_{i+1/2}$; u_{i+1} и u_i – значение температуры в узлах расчётной сетки x_{i+1} и x_i , соответственно; λ – параметр, имеющий смысл тепловой проводимости среды на участке $x_i \leq x < x_{i+1}$:

$$\lambda_i = \frac{k_i \cdot k_{i+1}}{k_i + k_{i+1}} \cdot \frac{2}{x_{i+1} - x_i}$$

Аналогично, рассматривая участок $[x_{i-1}, x_i]$, можно получить выражение для теплового потока во вспомогательном узле $x_{i-1/2}$:

$$w_{i-1/2} = -\lambda_{i-1} \cdot (u_i - u_{i-1}) \quad (59)$$

Приближенное значение интегрального (суммарного) конвекционного теплового потока через боковую поверхность контрольной области $[x_{i-1/2}, x_{i+1/2}]$:

$$\int_{x_{i-1/2}}^{x_{i+1/2}} r(x) \cdot u(x) \, dx \approx u(x_i) \int_{x_{i-1/2}}^{x_{i+1/2}} r(x) \, dx = \sigma_i \cdot u_i,$$

где σ – параметр имеющий смысл проводимости контактной области между средой и окружением:

$$\sigma_i = \int_{x_{i-1/2}}^{x_{i+1/2}} r(x) \, dx.$$

Для определения контактной проводимости среды σ_i используем кусочную аппроксимацию (рисунок 47), а также условие расположения узлов сетки $\{x_i\}$, таких что коэффициент конвекционного теплообмена $r(x)$ имеет *точки разрыва только в узлах расчётной сетки*:

$$\begin{aligned} \sigma_i &= \int_{x_{i-1/2}}^{x_i} r(x) \, dx + \int_{x_i}^{x_{i+1/2}} r(x) \, dx \approx \\ &\approx r(x_i) \cdot (x_i - x_{i-1/2}) + r(x_{i+1}) \cdot (x_{i+1/2} - x_i) = \\ &= r_i \cdot \frac{x_i - x_{i-1}}{2} + r_{i+1} \cdot \frac{x_{i+1} - x_i}{2}. \end{aligned} \tag{60}$$

Суммарное количество тепла, выделяемого распределенными источниками в пределах контрольной области $[x_{i-1/2}, x_{i+1/2}]$, определяется аналогично:

$$\begin{aligned} \varepsilon(x_i) &= \int_{x_{i-1/2}}^{x_{i+1/2}} q(x) \, dx = \int_{x_{i-1/2}}^{x_i} q(x) \, dx + \int_{x_i}^{x_{i+1/2}} q(x) \, dx \approx \\ &\approx q_i \cdot \frac{x_i - x_{i-1}}{2} + q_{i+1} \cdot \frac{x_{i+1} - x_i}{2}. \end{aligned} \tag{61}$$

Таким образом, баланс тепловой энергии в контрольной области $[x_{i-1/2}, x_{i+1/2}]$ с учетом соотношений для переноса тепла за счет теплопроводности материала среды (58), конвекционного теплообмена с окружающей средой (60) и распределенных источников (61), можно записать в виде системы

разностных уравнений:

$$\begin{cases} w_{i+1/2} - w_{i-1/2} + \sigma_i \cdot u_i = \varepsilon_i \\ w_{i+1/2} = -\lambda_i \cdot (u_{i+1} - u_i) \\ w_{i-1/2} = -\lambda_{i-1} \cdot (u_i - u_{i-1}) \end{cases} \quad (62)$$

Система уравнений (62) по своему построению является разностным аналогом основного дифференциального уравнения (50). Записывая уравнение (62) во всех узлах сетки, в которых оно определено ($i = 1, 2, \dots, n - 1$), получим систему из $2n - 1$ линейных алгебраических уравнений относительно $2n + 1$ неизвестных:

$$\begin{cases} u_0, & u_1, & \dots & u_i, & \dots & u_n, \\ w_{1/2}, & \dots & w_{i-1/2}, & w_{i+1/2}, & \dots & w_{n-1/2}, \end{cases} \quad (63)$$

Два недостающих уравнения задаются с помощью *аппроксимации краевых условий* (50).

9.2.4 Разностная аппроксимация краевых условий

Воспользуемся *интегро-интерполяционным методом* и проинтегрируем основное уравнение (50) вблизи левой границы рассматриваемой области $x = 0$:

$$\int_{x_0}^{x_{1/2}} w'(x) \, dx + \int_{x_0}^{x_{1/2}} r(x) \cdot u(x) \, dx = \int_{x_0}^{x_{1/2}} q(x) \, dx$$

Уравнение теплового баланса вблизи левой границы области $x = x_0$:

$$w_{1/2} - w_0 + \sigma_0 \cdot u_0 = \varepsilon_0, \quad (64)$$

где $w_{1/2}$ – тепловой поток в первом промежуточном узле $x_{1/2}$:

$$w_{1/2} = -\lambda_0 \cdot (u_1 - u_0); \quad (65)$$

w_0 – тепловой поток на левой границе области (50):

$$w_0 \equiv w(0) = s_0 - p_0 \cdot u(0) = s_0 - p_0 \cdot u_0; \quad (66)$$

σ_0 и ε_0 – контактная проводимость и количество тепла выделяемое распределенными источникам на левой границе:

$$\sigma_0 = \int_{x_0}^{x_{1/2}} r(x) dx \approx r(x_0) \cdot (x_{1/2} - x_0) = r_0 \cdot \frac{x_1 - x_0}{2},$$

$$\varepsilon_0 = \int_{x_0}^{x_{1/2}} q(x) dx \approx q(x_0) \cdot (x_{1/2} - x_0) = q_0 \cdot \frac{x_1 - x_0}{2}$$

Аналогично рассматривая правую границу области $x = \ell$, уравнение теплового баланса можно записать в виде:

$$w_n - w_{n-1/2} + \sigma_n \cdot u_n = \varepsilon_n, \quad (67)$$

где w_n – тепловой поток на правой границе области (50):

$$w_n \equiv w(\ell) = p_\ell \cdot u(\ell) - s_\ell = p_\ell \cdot u_n - s_\ell; \quad (68)$$

$w_{n-1/2}$ – тепловой поток в последнем промежуточном узле $x_{n-1/2}$:

$$w_{n-1/2} = -\lambda_{n-1} \cdot (u_n - u_{n-1}); \quad (69)$$

σ_n и ε_n – контактная проводимость и количество тепла выделяемое распределенными источникам на правой границе $x = \ell$:

$$\sigma_n = \int_{x_{n-1/2}}^{x_n} r(x) dx \approx r(x_n) \cdot (x_n - x_{n-1/2}) = r_n \cdot \frac{x_n - x_{n-1}}{2},$$

$$\varepsilon_n = \int_{x_{n-1/2}}^{x_n} q(x) dx \approx q(x_n) \cdot (x_n - x_{n-1/2}) = q_n \cdot \frac{x_n - x_{n-1}}{2}.$$

Объединяя все разностные соотношения для дифференциального уравнения (62) и граничных условия (64), (67) получаем следующую разностную

схему для граничной задачи:

$$\begin{cases} w_{1/2} - w_0 + \sigma_0 \cdot u_0 = \varepsilon_0 \\ w_{i+1/2} - w_{i-1/2} + \sigma_i \cdot u_i = \varepsilon_i \\ w_{i+1/2} = -\lambda_i \cdot (u_{i+1} - u_i) \\ w_{i-1/2} = -\lambda_{i-1} \cdot (u_i - u_{i-1}) \\ w_n - w_{n-1/2} + \sigma_n \cdot u_n = \varepsilon_n \end{cases} \quad (70)$$

Блок-схема алгоритма вычислений коэффициентов разностной аппроксимации краевой задачи представлена на рисунке 48.

Структура системы линейных уравнений (70) относительно неизвестных (63) позволяет сформулировать систему уравнений только для неизвестных $\{u_0, u_1, \dots, u_n\}$.

Подставим выражение для потоков в узлах вспомогательной сетки

$$\begin{cases} w_{i+1/2} = -\lambda_i \cdot (u_{i+1} - u_i) \\ w_{i-1/2} = -\lambda_{i-1} \cdot (u_i - u_{i-1}) \end{cases}$$

в уравнение теплового баланса для каждого контрольного объема

$$w_{i+1/2} - w_{i-1/2} + \sigma_i \cdot u_i = \varepsilon_i, \quad i = 1, 2, \dots, (n-1)$$

и в результате с учетом (66) и (68) получим:

$$\begin{cases} -\lambda_0 \cdot (u_1 - u_0) - (s_0 - p_0 \cdot u_0) + \sigma_0 \cdot u_0 = \varepsilon_0 \\ -\lambda_i \cdot (u_{i+1} - u_i) + \lambda_{i-1} \cdot (u_i - u_{i-1}) + \sigma_i \cdot u_i = \varepsilon_i \\ (p_\ell \cdot u_n - s_\ell) + \lambda_{n-1} \cdot (u_n - u_{n-1}) + \sigma_n \cdot u_n = \varepsilon_n \end{cases}$$

Группируя слагаемые при неизвестных u_{i-1}, u_i, u_{i+1} :

$$\begin{cases} c_0 \cdot u_0 - b_0 \cdot u_1 = y_0 \\ -a_i \cdot u_{i-1} + c_i \cdot u_i - b_i \cdot u_{i+1} = y_i, \quad i = 1, 2, \dots, n-1 \\ -a_n \cdot u_{n-1} + c_n \cdot u_n = y_n \end{cases} \quad (71)$$

где a_i , b_i и c_i – коэффициенты системы уравнений:

$$\begin{cases} b_0 = \lambda_0, & c_0 = b_0 + \sigma_0 + p_0, & y_0 = \varepsilon_0 + s_0 \\ a_i = \lambda_{i-1}, & b_i = \lambda_i, & c_i = a_i + b_i + \sigma_i, & y_i = \varepsilon_i \\ a_n = \lambda_{n-1}, & & c_n = a_n + \sigma_n + p_\ell, & y_n = \varepsilon_n + s_\ell \end{cases}$$

Таким образом, для нахождения неизвестных $\{u_0, u_1, \dots, u_n\}$ краевой задачи необходимо решить систему линейных уравнений (71), которую можно записать в матричном виде:

$$\mathbf{M} \cdot \mathbf{u} = \mathbf{y},$$

где $\mathbf{u} = (u_0, u_1, \dots, u_n)^T$ – вектор неизвестных, $\mathbf{y} = (y_0, y_1, \dots, y_n)^T$ – вектор правых частей системы, \mathbf{M} – матрица системы линейных уравнений:

$$\mathbf{M} = \begin{pmatrix} c_0 & -b_0 & 0 & & & & \\ -a_1 & c_1 & -b_1 & 0 & & & \\ 0 & -a_2 & c_2 & -b_2 & 0 & & \\ & 0 & \dots & \dots & \dots & 0 & \\ & & 0 & -a_i & c_i & -b_i & 0 \\ & & & 0 & \dots & \dots & \dots & 0 \\ & & & & 0 & -a_{n-1} & c_{n-1} & -b_{n-1} \\ & & & & & 0 & -a_n & c_n \end{pmatrix} \quad (72)$$

Следует отметить, что матрица \mathbf{M} содержит много нулевых элементов, расположенных в матрице не беспорядочно, а плотными массивами на заранее известных местах. Поэтому расчет по методу Гаусса можно организовать так, чтобы не задействовать при вычислениях нулевые элементы матрицы.

Поиск решения системы можно проводить стандартными методами линейной алгебры, хотя выбор наибольшего элемента в таких расчетах делать нельзя, так как перестановка столбцов разрушает специальную структуру матрицы. Кроме того, в матрицах с симметричной структурой недопустим выбор главного элемента матрицы.

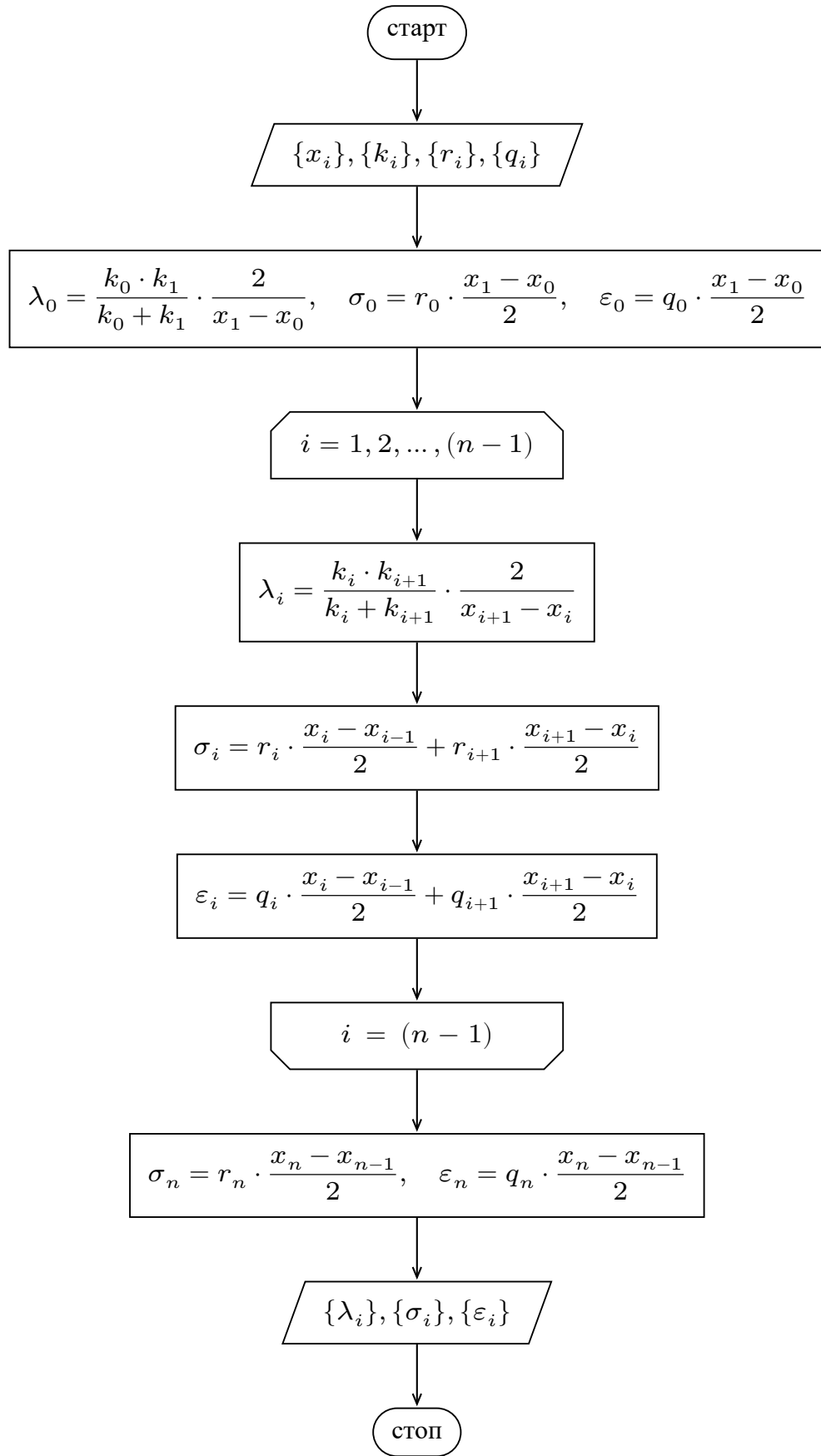


Рисунок 48 – Блок-схема алгоритма вычислений коэффициентов разностной схемы $\{\lambda_i\}, \{\sigma_i\}, \{\epsilon_i\}$

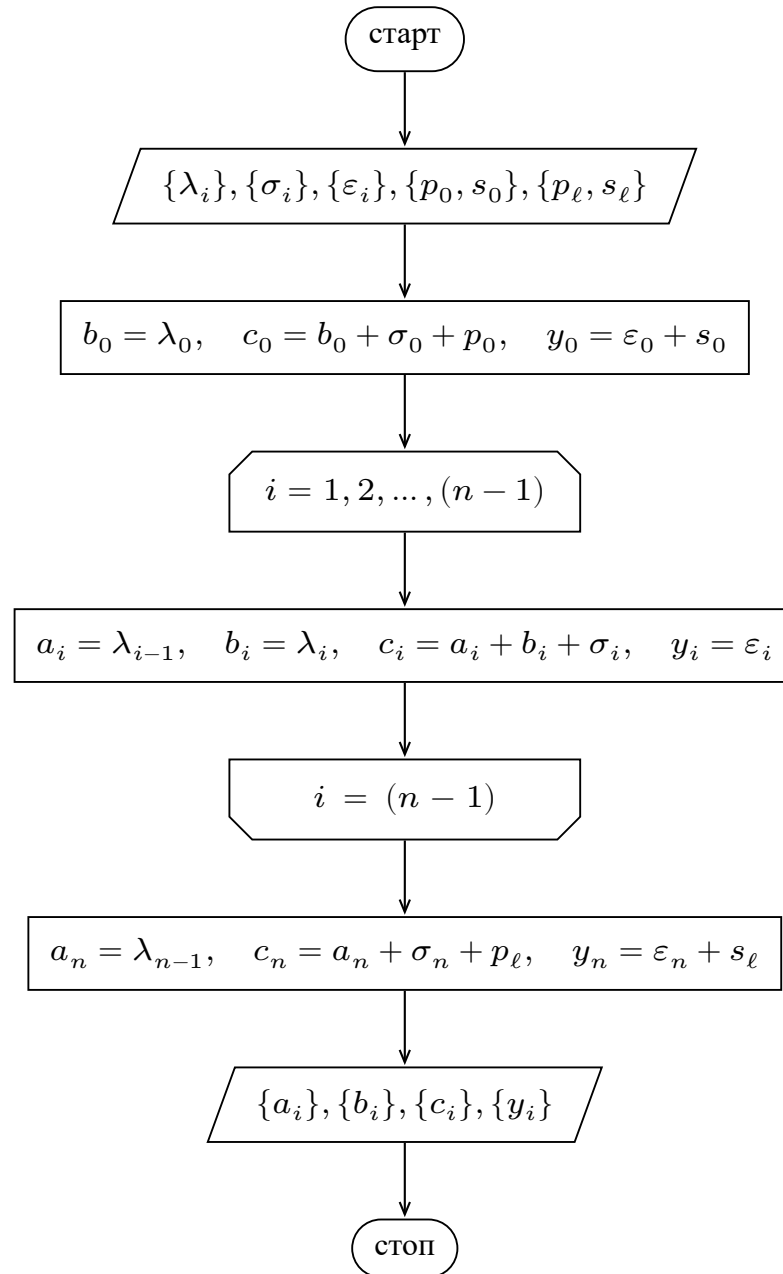


Рисунок 49 – Блок-схема алгоритма вычислений коэффициентов $\{a_i\}, \{b_i\}, \{c_i\}$ и $\{y_i\}$ системы линейных уравнений

9.3 Метод решения систем линейных уравнений с матрицами специального вида

9.3.1 Метод прогонки

Система уравнений (71) представляет собой частный случай систем линейных алгебраических уравнений с трехдиагональной матрицей \mathbf{M} , то есть с матрицей, все элементы которой, не лежащие на главной и двух побочных диа-

гоналях, равны нулю. Такие разреженные матрицы специального вида часто встречаются при решениях краевых задач для дифференциальных уравнений второго порядка.

Для численного решения таких систем уравнений применяется *метод прогонки*, который представляет собой вариант *метода Гаусса* последовательного исключения неизвестных. Основная идея метода прогонки заключается в предположении о наличии линейной связи между значениями «соседних» неизвестных и решение системы (71) ищется в виде:

$$u_i = \alpha_{i+1} \cdot u_{i+1} + \beta_{i+1}, \quad (73)$$

где α_{i+1} и β_{i+1} – неизвестные коэффициенты прогонки, которые необходимо определить.

Пользуясь уравнением (73) выразим u_{i-1} :

$$u_{i-1} = \alpha_i \cdot u_i + \beta_i, \quad \rightarrow \quad u_{i-1} = \alpha_i \alpha_{i+1} \cdot u_{i+1} + (\alpha_i \beta_{i+1} + \beta_i). \quad (74)$$

Подставляя соотношения для u_{i-1} (74) и u_i (73) в систему (71) получим:

$$-a_i \cdot (\alpha_i \alpha_{i+1} \cdot u_{i+1} + (\alpha_i \beta_{i+1} + \beta_i)) + c_i \cdot (\alpha_{i+1} \cdot u_{i+1} + \beta_{i+1}) - b_i \cdot u_{i+1} = y_i$$

Это уравнение будет выполнено для всех $i = 1, 2, \dots, n-1$, если потребовать равенство нулю всех коэффициентов при неизвестных u_{i+1} и свободных членов:

$$\begin{cases} (c_i - a_i \cdot \alpha_i) \cdot \alpha_{i+1} = b_i \\ (c_i - a_i \cdot \alpha_i) \cdot \beta_{i+1} = a_i \cdot \beta_i + y_i \end{cases}$$

Из последнего выражения получаем рекуррентные соотношения для определения значений неизвестных коэффициентов прогонки α_{i+1} и β_{i+1} :

$$\alpha_{i+1} = \frac{b_i}{c_i - a_i \cdot \alpha_i}, \quad \beta_{i+1} = \frac{a_i \cdot \beta_i + y_i}{c_i - a_i \cdot \alpha_i} \quad (75)$$

Соотношения этих коэффициентов представляют собой нелинейные разностные уравнения первого порядка, для решения которых необходимо задать начальные значения α_1 и β_1 , определяемые из краевого условия на левой гра-

нице рассматриваемой области $x = 0$:

$$\begin{cases} c_0 \cdot u_0 - b_0 \cdot u_1 = y_0 \\ u_0 = \alpha_1 \cdot u_1 + \beta_1 \end{cases} \quad \text{или} \quad \begin{cases} u_0 = \frac{b_0}{c_0} \cdot u_1 + \frac{y_0}{c_0} \\ u_0 = \alpha_1 \cdot u_1 + \beta_1 \end{cases}.$$

Сопоставляя выражение для u_0 получим:

$$\alpha_1 = \frac{b_0}{c_0}, \quad \beta_1 = \frac{y_0}{c_0}$$

Процесс вычисления коэффициентов α_{i+1} и β_{i+1} по рекуррентным соотношениям (75) называется *прямой прогонкой* (рисунок 50):

$$\begin{cases} \alpha_1 = \frac{b_0}{c_0}, & \alpha_{i+1} = \frac{b_i}{c_i - a_i \cdot \alpha_i} \\ \beta_1 = \frac{y_0}{c_0}, & \beta_{i+1} = \frac{a_i \cdot \beta_i + y_i}{c_i - a_i \cdot \alpha_i} \end{cases}, \quad i = 1, 2, \dots, (n-1) \quad (76)$$

После того как коэффициенты прогонки найдены, решение системы находится по рекуррентному соотношению (73), если известно значение функции u_n , которое можно определить из краевого условия на правой границе области $x = \ell$:

$$\begin{cases} -a_n \cdot u_{n-1} + c_n \cdot u_n = y_n \\ u_{n-1} = \alpha_n \cdot u_n + \beta_n \end{cases}. \quad (77)$$

Из решения системы уравнений (77) определяется значение неизвестной функции u_n на правой границе:

$$u_n = \frac{a_n \cdot \beta_n + y_n}{c_n - a_n \cdot \alpha_n}$$

Последовательное нахождение значений неизвестной функции $u(x)$ в узлах сетки по рекуррентному соотношению (73) называется *обратной прогонкой*:

$$u_n = \frac{a_n \cdot \beta_n + y_n}{c_n - a_n \cdot \alpha_n}, \quad u_i = \alpha_{i+1} \cdot u_{i+1} + \beta_{i+1}, \quad i = (n-1), \dots, 1, 0 \quad (78)$$

Неизвестные величины потоков $w_{i+1/2}$ можно определить рассматривая уравнение теплового баланса (62) как рекуррентное соотношение, выражающее

поток в одном промежуточном узле через поток в соседнем узле:

$$\begin{cases} w_0 = s_0 - p_0 \cdot u_0 \\ w_{1/2} = w_0 - \sigma_0 \cdot u_0 + \varepsilon_0 \\ w_{i+1/2} = w_{i-1/2} - \sigma_i \cdot u_i + \varepsilon_i, \quad i = 1, 2, \dots, (n-1) \\ w_n = w_{n-1/2} - \sigma_n \cdot u_n + \varepsilon_n \end{cases} \quad (79)$$

Алгоритм решения смешанной краевой задачи для системы обыкновенных дифференциальных уравнений методом прогонки:

- 1) Сформировать локально однородную расчётную сетку $\{x_i\}$ в пределах области поиска решений краевой задачи.
- 2) Вычислить коэффициенты разностной схемы $\{\lambda_i\}$, $\{\sigma_i\}$, $\{\varepsilon_i\}$ (рисунок 48).
- 3) Определить коэффициенты $\{a_i\}$, $\{b_i\}$, $\{c_i\}$ и $\{y_i\}$ системы линейной уравнений (рисунок 49).
- 4) Вычислить коэффициенты прогонки $\{\alpha_i\}$ и $\{\beta_i\}$ (рисунок 50)
- 5) Найти решение полученной системы линейных алгебраических уравнений $\{u_0, u_1, \dots, u_n\}$ (рисунок 51).
- 6) Вычислить поток неизвестной функции $\{w_0, w_{i\pm 1/2}, w_n\}$ во всех промежуточных узлах расчётной сетки (рисунок 52).

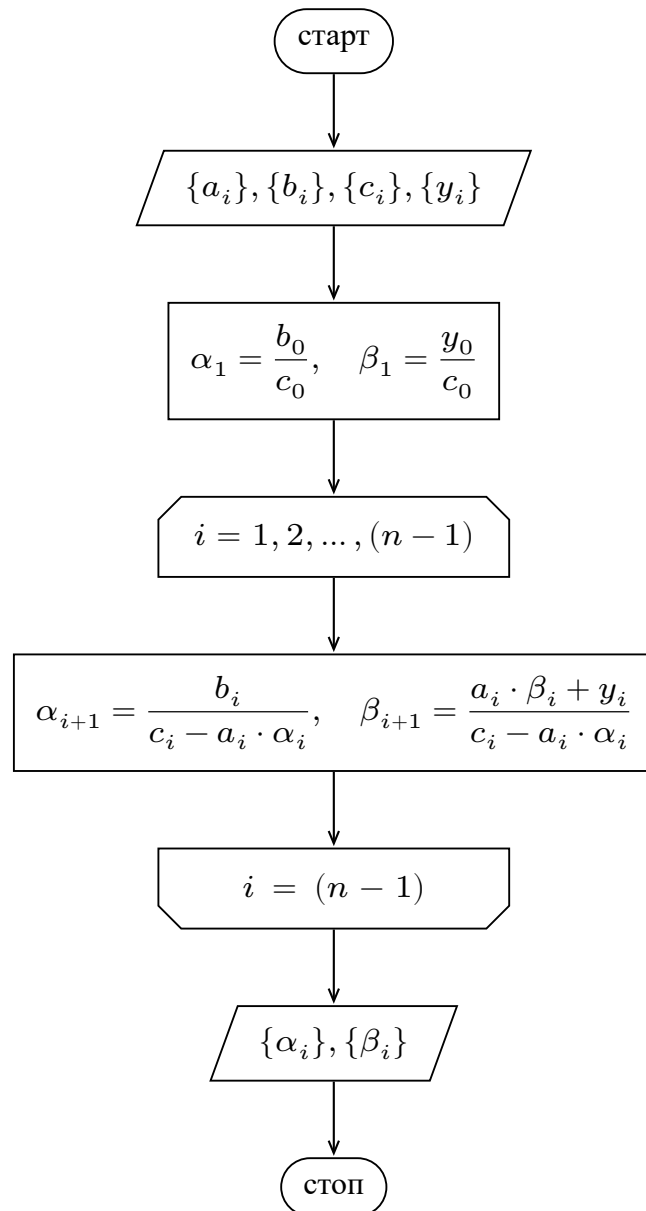


Рисунок 50 – Блок-схема алгоритма вычислений коэффициентов прогонки $\{\alpha_1, \alpha_2, \dots, \alpha_n\}$ и $\{\beta_1, \beta_2, \dots, \beta_n\}$

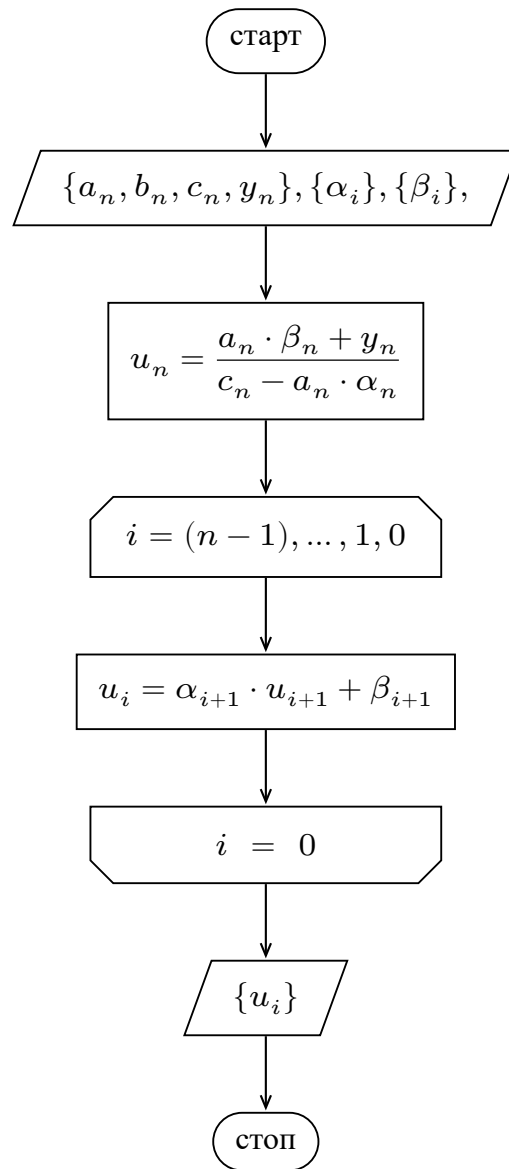


Рисунок 51 – Блок-схема алгоритма вычислений значений неизвестной функции $\{u_0, u_1, \dots, u_n\}$ в узлах расчетной сетки $\{x_0, x_1, \dots, x_n\}$

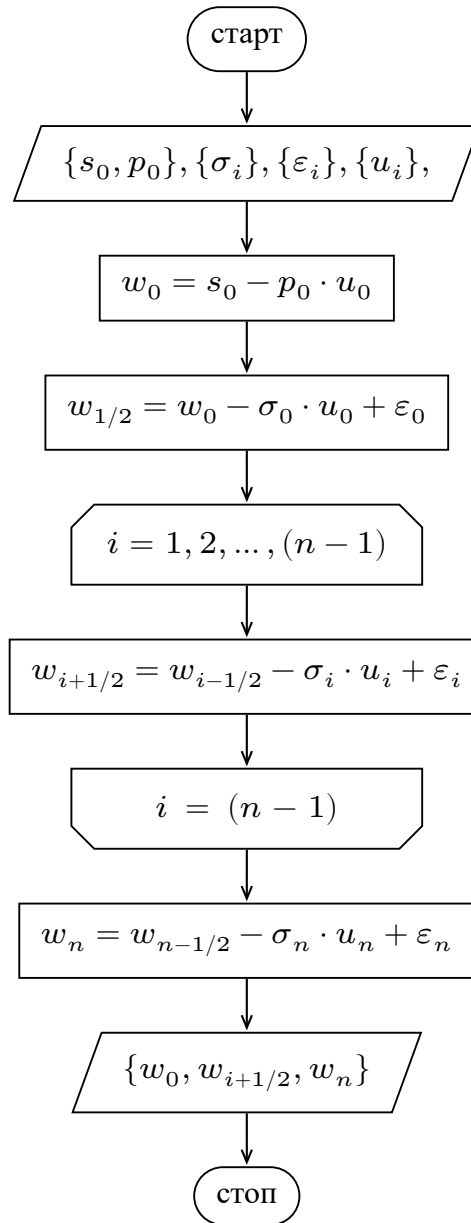


Рисунок 52 – Блок-схема вычислений потока неизвестной функции $\{w_0, w_{1/2}, w_{3/2}, \dots, w_{n-1/2}, w_n\}$ в промежуточных узлах расчетной сетки $\{x_0, x_{1/2}, x_{3/2}, \dots, x_{n-1/2}, x_n\}$