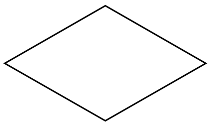
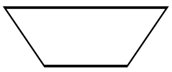
Решение любой задачи на ЭВМ состоит из следующих этапов:
1) математической постановки задачи;
2) выбора или разработки численного метода, изображения алгоритма в виде блок-схемы;
3) составления программы расчета на одном из языков программирования;
4) отладки программы, которая заключается в проведении проверочного варианта задачи, для которого известен результат, с последующим сравнением результатов;
5) проведения расчетов, анализа результатов.

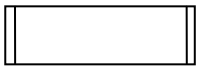
Под вычислительным алгоритмом понимают последовательность арифметических и логических операций, при помощи которых находится решение математической задачи, сформулированной на первом этапе. В пособии дается краткое изложение численных методов, приводятся некоторые рекомендации при практической реализации их на ЭВМ, а также имеются задания по каждой теме и графические описания некоторых алгоритмов в виде блок-схем, которые являются наглядным способом изображения решения задачи. Подробная блок-схема значительно облегчает процесс программирования и позволяет составить программу на любом алгоритмическом языке.

В схеме алгоритма каждому типу действий (вводу исходных данных, вычислению, проверке условий, управлению циклами, выводу результатов, окончанию счета) соответствует геометрическая фигура, представленная в виде блочного символа (блока). Представим наиболее часто употребляемые символы.


– начало или конец алгоритма

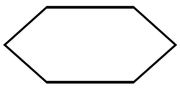

– проверка условия


– исходные данные


– вычисление по подпрограмме


– вычислительный процесс


– вывод результатов


– начало цикла

Блоки соединяются линиями со стрелками, указывающими последовательность выполнения действий. Алгоритм состоит из типовых структурных элементов, которые имеют линейную, разветвленную или циклическую структуру. *Алгоритм разветвленной структуры* – алгоритм, в котором предусмотрено разветвление выполняемой последовательности действий в зависимости от результата какого-либо условия. *Алгоритм* *циклической структуры* – алгоритм, в котором предусмотрено неоднократное выполнение одной и той же последовательности действий. Циклы делятся на *арифметические* (число повторений цикла заранее известно) и *итерационные* (число повторений цикла зависит от выполнения некоторого условия).

П р и м е р 1. Алгоритм циклической структуры (итерационный цикл). Вычислить с точностью ε

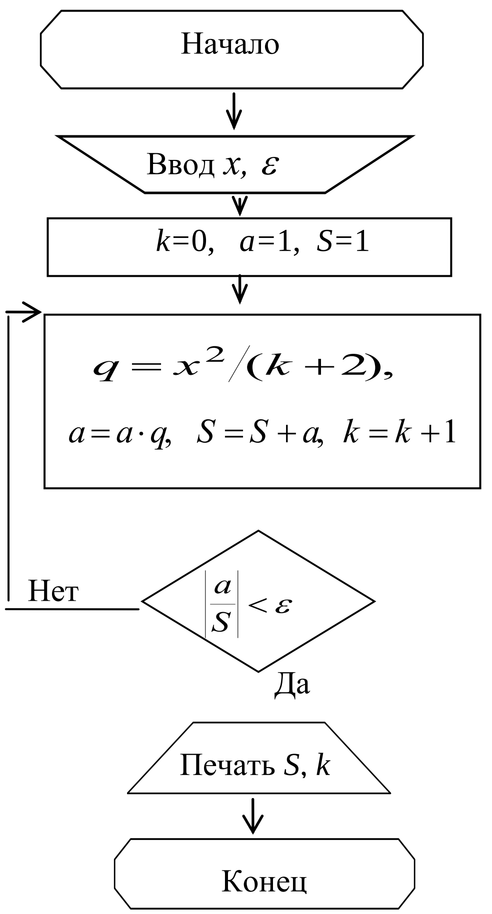
$$f(x)=\sum_{k=0}^{\infty}\frac{x^{2k}}{(k+1)!}=1+\frac{x^2}{2!}+\ldots+\frac{x^{2k}}{(k+1)!}+\frac{x^{2(k+1)}}{(k+2)!}+\ldots=\sum_{k=0}^{\infty}a_k(x),$$

где $a_k(x)=\dfrac{x^{2k}}{(k+1)!}$, $a_{k+1}(x)=a_k(x)\cdot q=\dfrac{x^{2k}}{(k+1)!}\cdot\dfrac{x^2}{k+2}$,

т.е. $q=\dfrac{x^2}{k+2}$, $S_k(x)=\sum\limits_{i=0}^{k}a_i(x)$.

Условием окончания счета будет $\left|\dfrac{S_{k+1}(x)-S_k(x)}{S_{k+1}(x)}\right|=\left|\dfrac{a_{k+1}(x)}{S_{k+1}(x)}\right|<\varepsilon.$

Блок-схема вычисления $f(x)$ имеет вид:

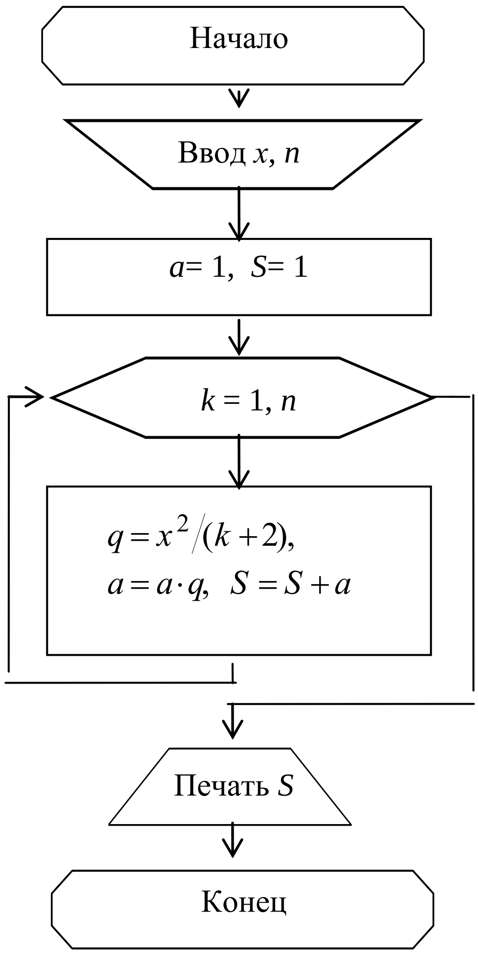


Исходные данные x и ε.
Здесь число повторений итерационного цикла зависит от заданной точности ε, чем меньше ε, тем больше k.
Результат вычислений $S=f(x)$.

П р и м е р 2. Алгоритм циклической структуры (арифметический цикл).

Вычислить $f(x)=\sum_{k=0}^{n}\frac{x^{2k}}{(k+1)!}$

Блок-схема вычисления $f(x)$ имеет вид:



Исходные данные x и n.
Здесь число повторений цикла заранее известно, равно n.
Результат вычислений $S=f(x)$.

Пособие включает следующие темы: интерполяция и приближение функции, численное дифференцирование и интегрирование, численное решение систем линейных алгебраических уравнений, решение нелинейных уравнений, численные методы решения задачи Коши, разностный метод решения краевых задач для обыкновенных дифференциальных уравнений.

1. ИНТЕРПОЛЯЦИЯ И ПРИБЛИЖЕНИЕ ФУНКЦИИ

1.1. Постановка задачи

Часто требуется восстановить функцию $y(x)$ для всех значений x на $[a,b]$, если известны ее значения в некотором конечном числе точек этого отрезка. Пусть на $[a,b]$ задана сетка $\overline{\omega} = \{x_0 = a < x_1 < \ldots < x_n = b\}$, и в ее узлах заданы значения функции $y(x)$, $y_i = y(x_i),\ i = 0,1,\ldots,n.$

Требуется построить интерполянту – функцию $P(x)$, совпадающую с функцией $y(x)$ в узлах сетки

$$P(x_i) = y_i, \quad i = 0,1,\ldots,n. \tag{1}$$

Схема решения задачи сводится к следующему: строится достаточно простая функция $P(x)$ (например, многочлен), которая «близка» к $y(x)$ в пределах заданной точности ε, и приближенно полагается $y(x) \approx P(x)$. Величина $\varepsilon(x) = y(x) - P(x)$ представляет собой погрешность метода приближения.

1.2. Полиномиальная интерполяция

Известно, что любая непрерывная на $[a,b]$ функция $y(x)$ может быть хорошо приближена некоторым полиномом $P_n(x) = \sum_{k=0}^{n} C_k x^k$.

ТЕОРЕМА Вейерштрасса. Для любого $\varepsilon > 0$ существует полином $P_n(x)$ степени $n = n(\varepsilon)$ такой, что $\max_{a \le x \le b} \left| y(x) - P_n(x) \right| < \varepsilon$.

Частный случай полиномиальной интерполяции – кусочно-линейная интерполяция, т.е. когда $n = 1$. Заменим функцию $y(x)$ на каждом отрезке $[x_i, x_{i+1}]$ (i=0,1,...,n-1) на линейную $P_1(x) = C_0 + C_1 x$, где $x_i \le x \le x_{i+1}$. Соединим точки графика , (x_i, y_i), i=0,1,...,n, ломаной линией. Полученная кусочно-линейная функция будет искомым приближением функции $y(x)$.

Из условия (1) $P_1(x_i)=y_i,\quad i=0,1,\ldots,n$, находим коэффициенты C_0 и C_1 для каждого отрезка $[x_i,x_{i+1}]$ (i=0,1,...,n-1)

$$\begin{cases} C_0+C_1x_i=y_i \\ C_0+C_1x_{i+1}=y_{i+1} \end{cases} \Rightarrow C_1=\frac{y_{i+1}-y_i}{x_{i+1}-x_i},\quad C_0=y_i-\frac{y_{i+1}-y_i}{x_{i+1}-x_i}\cdot x_i. \qquad (2)$$

Тогда $P_1(x)=y_i+\dfrac{y_{i+1}-y_i}{x_{i+1}-x_i}\cdot(x-x_i)$, где $x_i\le x\le x_{i+1}$.

Таким образом, $P_1(x)$ является многочленом первой степени, и $y(x)=P_1(x)+\varepsilon_1(x)$, причем, если $y(x)\in C^2[x_i,x_{i+1}]$, то $\varepsilon_1(x)=y''(\xi)(x-x_i)(x-x_{i+1})/2$, где $\xi\in[x_i,x_{i+1}]$.

1.3. Интерполяционный многочлен Лагранжа

Если линейная интерполяция оказывается недостаточно точной, то используют аппроксимацию функции $y(x)$ многочленом степени n>1. Многочлен $L_n(x)$ степени n такой, что

$$L_n(x_i)=y_i,\quad i=0,1,\ldots,n,$$

называется интерполяционным многочленом.

Известно, что интерполяционный многочлен существует, единственен и может быть записан в виде

$$L_n(x)=\sum_{i=0}^{n}y_il_{in}(x), \qquad (3)$$

где $l_{in}=\prod\limits_{\substack{k=0\\k\ne i}}^{n}\dfrac{x-x_k}{x_i-x_k}=\dfrac{\omega_n(x)}{\omega_n'(x_i)(x-x_i)},\qquad \omega_n(x)=(x-x_0)\ldots(x-x_n)$.

Многочлен $L_n(x)$ называется интерполяционным многочленом в форме Лагранжа. Итак, $y(x)=L_n(x)+\varepsilon_n(x)$, причем, если $y(x)\in C^{n+1}[a,b]$, то погрешность имеет вид $\varepsilon_n(x)=\omega_n(x)f^{(n+1)}(\xi)/(n+1)!,\ \xi\in[a,b]$.

1.4. Сплайн-интерполяция

Рассмотрим специальный случай кусочно-полиномиальной интерполяции, когда между любыми соседними узлами сетки функция $y(x)$ интерполируется кубическим полиномом

$$S(x)=a_i+b_i(x-x_{i-1})+c_i(x-x_{i-1})^2+d_i(x-x_{i-1})^3, \qquad (4)$$
$$x_{i-1}\le x\le x_i.$$

Его коэффициенты $a_i,\ b_i,\ c_i,\ d_i$ на каждом интервале $[x_{i-1},x_i]$ определяются из условий сопряжения в узлах

$$
\begin{aligned}
&S(x_i) = y_i,\\
&S'(x_i - 0) = S'(x_i + 0),\\
&S''(x_i - 0) = S''(x_i + 0), \quad i = 1,2,\ldots,n-1.
\end{aligned} \tag{5}
$$

Кроме того, на границе при $x=x_0$ и $x=x_n$ ставится условие

$$S''(x_0) = 0, \quad S''(x_n) = 0. \tag{6}$$

Можно ввести понятие сплайна порядка m как функции, которая является полиномом степени m на каждом из отрезков сетки и во всех внутренних узлах сетки удовлетворяет условиям непрерывности функции и производных до порядка m включительно. Обычно для интерполяции используются случаи $m=3$ (кубический сплайн) и $m=1$ (линейный сплайн), соответствующий аппроксимации графика функции $y(x)$ ломаной, проходящей через точки (x_i, y_i).

2. ЧИСЛЕННОЕ ДИФФЕРЕНЦИРОВАНИЕ И ИНТЕГРИРОВАНИЕ

2.1. Простейшие разностные операторы

Пусть $y(x)$ –достаточно гладкая функция, и нам известны значения функции $y_i=y(x_i)$ для всех узлов x_i , принадлежащих сетке
$\overline{\omega_h} = \{x_i = a + ih, \ i = 0,1,\ldots,n, \quad h = (b-a)/n\}$.

Вместо функции непрерывного аргумента $x \in [a,b]$ рассматривается функция $y(x_i) = y_i$ дискретного аргумента $x_i \in \overline{\omega}_h$, которая называется сеточной. Любую сеточную функцию $y(x_i) = y_i$ можно представить в виде вектора $Y = (y_0, y_1, \ldots, y_n)$. Поэтому множество сеточных функций образует конечномерное пространство H_h, в данном случае размерности $n+1$. Обычно рассматривается семейство сеток $\{\overline{\omega}_h\}$, зависящих от шага h как от параметра (или от n). В пространстве H_h можно ввести норму. Простейшие виды норм:

$$\| y \|_C = \max_{0 \le i \le n} |y_i|; \quad \| y \| = \left(\sum_{i=1}^{n-1} y_i^2 h \right)^{1/2}.$$

Дифференциальный оператор $Ly = y'(x) = \dfrac{dy}{dx}$ заменяется разностным оператором $L_h y$, действующим в пространстве сеточных функций,

$$(L_h y)_i = \sum_{x_j \in \sigma_i} a_{ij}^h y_h(x_j), \ x_i \in \omega_h, \ \sigma_i \in \omega_h,$$

где a_{ij}^h – коэффициенты, σ_i – шаблон.

Такая замена дифференциального оператора Ly на $L_h y$ называется аппроксимацией на сетке дифференциального оператора L разностным оператором L_h. Построение L_h надо начинать с выбора шаблона σ_i, т.е. множества узлов, соседних с узлом $x_i \in \omega_h$, в которых значения сеточной функции y_i могут быть использованы при написании выражения для L_h. Рассмотрим примеры построения L_h.

П р и м е р 1. Первая производная $Ly = \dfrac{dy}{dx} = y'(x)$. В качестве шаблона σ_i возьмем три узла $(x_i - h, x_i, x_i + h)$, т.е. (x_{i-1}, x_i, x_{i+1}). Можно воспользоваться любым из выражений:

$L_h^+ y = \dfrac{y(x_i + h) - y(x_i)}{h} = y_x$ – это правая разностная производная,

$L_h^- y = \dfrac{y(x_i) - y(x_i - h)}{h} = y_{\bar{x}}$ – левая разностная производная,

$L_h^0 y = \dfrac{y(x_i + h) - y(x_i - h)}{2h} = y_{\overset{\circ}{x}}$ – центральная разностная производная.

На трехточечном шаблоне $(x_i - h, x_i, x_i + h)$ можно определить разностный оператор $L_h^{(\sigma)} y = \sigma y_x + (1 - \sigma) y_{\bar{x}}$, где σ – действительный параметр. Таким образом, существует бесконечное множество разностных аппроксимаций первой производной на трехточечном шаблоне.

Погрешностью аппроксимации оператора L оператором L_h называют разность $\psi = L_h y - Ly$. Говорят, что L_h имеет m-й порядок аппроксимации в точке x, если $\psi = L_h y - Ly = O(h^m)$, или $|\psi(x)| \le M \cdot h^m$, где M=const>0 не зависит от h, m>0. Используя формулу Тейлора

$$y(x \pm h) = y(x) \pm h y'(x) + \frac{h^2}{2} y''(x) \pm \frac{h^3}{3!} y'''(x) + \frac{h^4}{4!} y^{\mathrm{iv}}(x) + O(h^5),$$

можно получить оценки:

$$\psi = y_x - y' = \frac{y(x+h) - y(x)}{h} - y'(x) = \frac{y(x) + h y'(x) + \frac{h^2}{2} y'' + \ldots - y(x)}{h} -$$
$$- y'(x) = O(h),$$

$$\psi = y_{\bar{x}} - y' = \frac{y(x) - y(x-h)}{h} - y'(x) = O(h),$$

$$\psi = y_{\overset{\circ}{x}} - y' = \frac{y(x+h) - y(x-h)}{2h} - y'(x) = O(h^2).$$

Таким образом, y_x и $y_{\bar{x}}$ имеют первый порядок аппроксимации, а $y_{\overset{\circ}{x}}$ – второй порядок аппроксимации.

П р и м е р 2. Вторая производная $Ly = \dfrac{d^2y}{dx^2} = y''(x)$. Возьмем тот же трехточечный шаблон, что и в примере 1, и напишем разностный оператор

$$L_h y(x) = \frac{y(x+h) - 2y(x) + y(x-h)}{h^2} = y_{\bar{x}x}.$$

Замечая, что $y(x+h) = y(x) + hy_x, \quad y(x-h) = y(x) - hy_{\bar{x}}$, преобразуем

$$L_h y(x) = \frac{y(x) + hy_x - 2y(x) + y(x) - hy_{\bar{x}}}{h^2} = \frac{y_x - y_{\bar{x}}}{h} = y_{\bar{x}x}(x).$$

Пользуясь формулой Тейлора для $y(x \pm h)$, находим

$$\psi = L_h y - Ly = y_{\bar{x}x} - y'' = \frac{h^2}{24} y^{\text{iv}}(x) + O(h^4) = O(h^2),$$

т.е. $y_{\bar{x}x}$ имеет второй порядок аппроксимации.

Обычно требуется оценка погрешности аппроксимации на сетке, т.е. в некоторой сеточной норме $\|\cdot\|_h$. Говорят, что L_h имеет m-й порядок аппроксимации на сетке, если $\|L_h y_h - (Ly)_h\|_h = O(h^m)$.

2.2. Численное интегрирование. Простейшие квадратурные формулы

Задача численного интегрирования состоит в нахождении приближенного значения интеграла

$$J(f) = \int_a^b f(x)dx, \tag{7}$$

где $f(x)$ – заданная функция.

На $[a,b]$ вводится сетка $\bar{\omega} = \{x_0 = a < x_1 < \ldots < x_N = b\}$, и в качестве приближенного значения интеграла рассматривается число

$$J_N(f) = \sum_{i=0}^{N} C_i f(x_i), \tag{8}$$

где $f(x_i)$ – значения функции $f(x)$ в узлах сетки $x = x_i$, C_i – коэффициенты, зависящие только от узлов, но не зависящие от $f(x)$.

Формула (8) называется квадратурной формулой. Задача численного интегрирования при помощи квадратур состоит в отыскании таких узлов $\{x_i\}$ и таких коэффициентов $\{C_i\}$, чтобы погрешность квадратурной формулы $R_N(f)=\sum_{i=0}^{N} C_i f(x_i) - \int_a^b f(x)dx = J_N(f) - J(f)$ была минимальной для функций из заданного класса.

Рассмотрим простейшие квадратурные формулы.

1. Формулы прямоугольников

$$J_N(f)=\sum_{i=0}^{N-1} hf(x_i),\quad J_N(f)=\sum_{i=1}^{N} hf(x_i),\quad J_N(f)=\sum_{i=0}^{N-1} hf(x_i+\frac{h}{2}), \tag{9}$$

где h – шаг в равномерной сетке

$\overline{\omega_h}=\{x_i=a+ih,\ i=0,1,\ldots,N,\ h=(b-a)/N\}$.

Погрешность формулы прямоугольников

$$|R_N(f)|\le \frac{h^2}{24}(b-a)M_2,\ \text{где}\ M_2=\max_{x\in[a,b]}|f''(x)|.$$

2. Формула трапеций

$$J_N(f)=h\big((f(x_0)+f(x_N))/2+f(x_1)+\ldots+f(x_{N-1})\big). \tag{10}$$

Погрешность формулы трапеций

$$|R_N(f)|\le \frac{h^2}{12}(b-a)M_2,\ \text{где}\ M_2=\max_{x\in[a,b]}|f''(x)|.$$

3. Формула Симпсона.

Метод Симпсона обеспечивает большую точность вычисления определенного интеграла и заключается в следующем. Кривая подинтегральной функции заменяется кусочно-непрерывной линией, состоящей из отрезков квадратичных парабол; весь интервал интегрирования разбивается на четное число N=2m отрезков. Формула Симпсона с постоянным шагом h имеет вид

$$J_{2m}(f)=\frac{h}{3}\big[f_0+f_{2m}+4(f_1+f_3+\ldots+f_{2m-1})+2(f_2+f_4+\ldots+f_{2m-2})\big], \tag{11}$$

где $f_0=f(a),\ f_{2m}=f(b),\ f_i=f(x_i),\ i=1,2,\ldots,2m-1.$

Погрешность формулы Симпсона

$$|R_{2m}(f)|\le \frac{b-a}{180}h^4\left|f^{(4)}(\xi)\right|,\ \text{где}\ \xi\in[a,b].$$

3. ЧИСЛЕННОЕ РЕШЕНИЕ СИСТЕМ ЛИНЕЙНЫХ

АЛГЕБРАИЧЕСКИХ УРАВНЕНИЙ (СЛАУ)

Численные методы решения системы $AX=B$, или

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

условно разделяют на две группы: прямые и итерационные. Прямые методы позволяют за конечное число действий получить точное решение системы уравнений, если вычисления ведутся без округления и коэффициенты матрицы заданы точно. Простейшие примеры прямых методов: **метод Гаусса**, **метод прогонки**. Итерационный метод позволяет найти приближенное решение системы с точностью $\varepsilon>0$ путем построения последовательности приближений, начиная с некоторого начального приближения. Само приближенное решение является результатом вычислений, полученным после конечного числа итераций $n(\varepsilon)$. Итерационные методы широко применяются для решения разностных уравнений математической физики, которым соответствуют ленточные матрицы A высокого порядка, ненулевые элементы которых находятся вблизи главной диагонали.

3.1. Метод Гаусса

Метод Гаусса относится к прямым методам решения СЛАУ и состоит из двух этапов: прямого и обратного хода.

Прямой ход. Система (12) приводится к треугольному виду $X+\overline{A}X=\overline{B}$, где $\overline{A}$ – верхняя треугольная наддиагональная матрица.

Предположим, что $a_{11}\neq 0$. Разделим первое уравнение в (12) на a_{11}. Из остальных исключим неизвестное x_1. Для этого из i-го уравнения вычитаем первое, умноженное на a_{i1}/a_{11}. Введем обозначение $b_i=a_{i,\,n+1}$ $(i=\overline{1,n})$.

Тогда система (12) преобразуется к виду

$$x_1+a_{12}^{(1)}x_2+\ldots+a_{1n}^{(1)}x_n=a_{1,n+1}^{(1)},$$

$$\left.\begin{aligned} a_{22}^{(1)}x_2+\ldots+a_{2n}^{(1)}x_n&=a_{2,n+1}^{(1)}, \\ \ldots\ldots\ldots\ldots\ldots\ldots\ldots&\ldots\ldots, \\ a_{n2}^{(1)}x_2+\ldots+a_{nn}^{(1)}x_n&=a_{n,n+1}^{(1)}, \end{aligned}\right\}. \qquad (13)$$

где $a_{ij}^{(1)}=a_{ij}-a_{i1}\cdot a_{1j}/a_{11},\quad j=\overline{2,n+1},\ i=\overline{2,n},\ a_{1j}^{(1)}=a_{1j}/a_{11}$.

Далее процесс повторяется применительно к системе (13). В итоге после выполнения n шагов получим систему с треугольной матрицей

$$\begin{aligned} x_1 + a_{12}^{(1)} x_2 + a_{13}^{(1)} x_3 + \ldots + a_{1n}^{(1)} x_n &= a_{1,n+1}^{(1)}, \\ x_2 + a_{23}^{(2)} x_3 + \ldots + a_{2n}^{(2)} x_n &= a_{2,n+1}^{(2)}, \\ \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots \\ x_n &= a_{n,n+1}^{(n)}. \end{aligned} \tag{14}$$

Попутно в прямом ходе метода Гаусса мы вычисляем определитель матрицы A

$$\det A = a_{11} \cdot a_{22}^{(1)} \cdot \ldots \cdot a_{nn}^{(n-1)}. \tag{15}$$

Обратный ход. Из соотношений (14) последовательно находим x_i, $i=n,n$-1,...,1 по рекуррентной формуле

$$\begin{aligned} x_n &= a_{n,n+1}^{(n)}, \\ x_i &= a_{i,n+1}^{(i)} - \left(a_{i,i+1}^{(i)} x_{i+1} + \ldots + a_{i,n}^{(i)} x_n \right), \quad i = n-1, \ldots, 1. \end{aligned} \tag{16}$$

3.2. Итерационные методы решения СЛАУ. Метод простой итерации

Для применения метода простой итерации необходимо СЛАУ $AX=B$ преобразовать к виду $X=CX+D$. Например, если $a_{ii} \neq 0$, таким образом:

$$x_i^{(k+1)} = \frac{1}{a_{ii}} \left(b_i - \sum_{j \neq i}^{1 \div n} a_{ij} x_j^{(k)} \right), \quad i = 1, 2, \ldots, n. \tag{17}$$

В качестве начального приближения можно брать любой вектор $X^{(0)} = (x_1^{(0)}, x_2^{(0)}, \ldots, x_n^{(0)})$, например, $X^{(0)} = B.$ По достижении заданной точности $\varepsilon > 0$ вычисления прекращаются. Условием окончания счета можно взять, например,

$$\max_{1 \le i \le n} \left| x_i^{(k+1)} - x_i^{(k)} \right| < \varepsilon, \quad \text{или} \tag{18}$$

$$\sqrt{\sum_{i=1}^{n} \left| x_i^{(k+1)} - x_i^{(k)} \right|^2} < \varepsilon. \tag{19}$$

3.3. Метод Зейделя

Модификацией метода простой итерации является метод Зейделя:

$$x_1^{(k+1)} = \frac{1}{a_{11}}\left(b_1 - \sum_{j=2}^{n} a_{1j} x_j^{(k)}\right), \tag{20}$$

$$x_i^{(k+1)} = \frac{1}{a_{ii}}\left(b_i - \sum_{j=1}^{i-1} a_{ij} x_j^{(k+1)} - \sum_{j=i+1}^{n} a_{ij} x_j^{(k)}\right), i = 2, \ldots, n. \tag{21}$$

В методе Зейделя при вычислении очередной координаты $x_i^{(k+1)}$ используются только что полученные координаты $x_1^{(k+1)}, \ldots, x_{i-1}^{(k+1)}$. Условием окончания счета можно взять (18) или (19).

3.4. Метод верхней релаксации

Ускорить итерационный процесс Зейделя можно с помощью итерационного параметра ω, в результате получим метод верхней релаксации:

$$x_i^{(k+1)} = x_i^{(k)} + \frac{\omega}{a_{ii}}\left(b_i - \sum_{j=1}^{i-1} a_{ij} x_j^{(k+1)} - \sum_{j=i}^{n} a_{ij} x_j^{(k)}\right), \quad i = 1, 2, \ldots, n. \tag{22}$$

При $\omega=1$ получаем формулу метода Зейделя. Для сходимости метода верхней релаксации надо потребовать, чтобы $1 \le \omega < 2$.

4. РЕШЕНИЕ НЕЛИНЕЙНЫХ УРАВНЕНИЙ. ИТЕРАЦИОННЫЕ МЕТОДЫ

Рассмотрим нелинейное уравнение

$$f(x) = 0, \quad x \in [a, b], \tag{23}$$

где $f(x)$ – непрерывная функция. Уравнение может иметь один или несколько корней, для нахождения которых применяются итерационные методы.

4.1. Метод дихотомии

Простейшим является метод дихотомии (деления пополам). Пусть $f(x_0) \cdot f(x_1) \le 0$, тогда на отрезке $[x_0, x_1]$ лежит не менее одного корня. Найдем $f(x_2)$, где $x_2 = (x_0 + x_1)/2$ и возьмем x_3 – то из значений x_0 или x_1, для которого выполняется условие $f(x_3) \cdot f(x_2) \le 0$. Отрезок

$\left[x_2, x_3\right]$ снова делим пополам и т.д. Деление продолжим до тех пор, пока длина отрезка станет меньше 2ε, где ε – точность, с которой требуется определить корень. Процесс сходится со скоростью геометрической прогрессии со знаменателем 1/2.

4.2. Метод простой итерации

Перепишем уравнение (23) в виде

$$x = \varphi(x), \tag{24}$$

где $\varphi(x)$ можно определить одним из способов:

$$\varphi(x) = x - \alpha f(x), \quad \alpha = const,$$

$$\varphi(x) = x + p(x) f(x),$$

$p(x)$ – произвольная функция, не имеющая корней на отрезке $[a, b]$.

Метод простой итерации определяется формулой

$$x_{n+1} = \varphi(x_n), \quad n = 0,1,2,\ldots, \tag{25}$$

где n – номер итерации, x_0 – заданное начальное приближение.

Требуется найти приближенно решение $x = x^*$ уравнения $x = \varphi(x)$ с относительной погрешностью $\varepsilon > 0$ так, чтобы при всех $n \geq n_0(\varepsilon)$ выполнялось неравенство

$$\left|x_n - x^*\right| \leq \varepsilon \left|x_0 - x^*\right|, \quad n \geq n_0(\varepsilon).$$

Это условие на практике может быть заменено следующим:

$$\left|\frac{x_{n+1} - x_n}{x_{n+1}}\right| \leq \varepsilon. \tag{26}$$

Метод простой итерации сходится со скоростью геометрической прогрессии со знаменателем $q < 1$.

4.3. Метод Ньютона

Метод Ньютона определяется формулой

$$x_{n+1} = x_n - f(x_n) / f'(x_n), \quad f'(x_n) \neq 0, \;\; n = 0,1,2,\ldots \tag{27}$$

Этот метод называют также методом касательных или методом линеаризации. Формула (27) получается, если в разложении

$$0 = f\left(x^*\right) = f(x_n) + f'(x_n)\left(x^* - x_n\right) + f''(\xi)\left(x^* - x_n\right)^2 / 2, \tag{28}$$

где $\xi = x_n + \Theta\left(x^* - x_n\right), \quad 0 \leq \Theta \leq 1,$

отбросить последний член, заменив x^* на x_{n+1}:

$$0 = f(x_n) + f'(x_n)(x_{n+1} - x_n).$$

Записывая $f(x) = 0$ в виде $x = \varphi(x)$, видим, что метод Ньютона можно трактовать как метод простой итерации с правой частью

$$\varphi(x) = x - f(x)/f'(x). \qquad (29)$$

Метод Ньютона сходится с квадратичной скоростью.

5. ЧИСЛЕННЫЕ МЕТОДЫ РЕШЕНИЯ ЗАДАЧИ КОШИ

5.1. Постановка задачи

Пусть требуется найти непрерывную при $t_0 \le t \le T$ функцию $u = u(t)$, удовлетворяющую дифференциальному уравнению при $t > t_0$

$$\frac{du}{dt} = f(t, u(t)), \quad t_0 < t \le T, \qquad (30)$$

и начальному условию $u(t_0) = u_0$, где $f(t, u)$ – заданная непрерывная функция двух аргументов.

Если функция $f(t, u)$ определена в прямоугольнике $D = \{t_0 \le t \le T,\ |u - u_0| \le U\}$ и удовлетворяет в области D по переменной u условию Липшица:

$$|f(t, u_1) - f(t, u_2)| \le K|u_1 - u_2| \qquad (31)$$

для всех $(t, u_1), (t, u_2) \in D$, где $K = const > 0$, то задача Коши (30) имеет единственное решение.

5.2. Разностная схема Эйлера

Простейшим численным методом решения уравнения (30) является разностная схема Эйлера. Введем на отрезке интегрирования $t_0 \le t \le T$ сетку $\bar{\omega}_\tau = \{t_n = t_0 + n\tau, \quad n = 0,1,\ldots,N, \quad \tau = (T - t_0)/N\}$. Будем обозначать через $y_n = y(t_n)$ сеточную функцию. Заменяя производную $\frac{du}{dt}$ в (30) на сетке ω_τ разностным отношением $y_{t,n} = \frac{y_{n+1} - y_n}{\tau}$, получим схему Эйлера:

$$\begin{aligned} &\frac{y_{n+1} - y_n}{\tau} = f(t_n, y_n), \quad n = 0,1,\ldots,N-1, \\ &y_0 = u_0. \end{aligned} \qquad (32)$$

Значения $y_n = y(t_n)$ находятся последовательно, начиная с $y_0 = u_0$, по явной формуле

$$y_{n+1} = y_n + \tau f(t_n, y_n), \quad n = 0,1,\ldots,N-1, \quad y_0 = u_0.$$

Вместо $u = u(t)$ мы находим сеточную функцию $y_n = y(t_n)$ − приближенное решение задачи (30). Сеточная функция $z_n = y_n - u(t_n)$ является погрешностью разностной схемы. Подставляя в (32) вместо $y_n = z_n + u_n$, получим уравнение для z_n

$$\frac{z_{n+1} - z_n}{\tau} = \alpha_n z_n + \psi_n, \quad n = 0,1,\ldots,N-1, \quad z_0 = 0, \tag{33}$$

где $\alpha_n = f_u(t_n, u_n + \Theta z_n), \quad 0 \le \Theta \le 1.$

ψ_n− невязка, или погрешность аппроксимации схемы (32) на решении $u = u(t)$ задачи (30), равна

$$\psi_n = f(t_n, u_n) - (u_{n+1} - u_n)/\tau. \tag{34}$$

Подставляя в (34)

$$u_{n+1} = u_n + \tau u_n' + \frac{\tau^2}{2} u_n'' + \ldots \tag{35}$$

и учитывая, что согласно (30) $u_n' = f(t_n, u_n)$, получим $\psi_n = O(\tau)$, или $\|\psi\|_C = \max\limits_{t_0 \le t_n \le T} |\psi_n| = O(\tau)$. Это означает, что схема Эйлера имеет первый порядок аппроксимации. Можно доказать, что схема Эйлера сходится и имеет первый порядок точности, т.е. $\|z\|_C = \max\limits_{t_0 \le t_n \le T} |z_n| = O(\tau)$. Для этого достаточно предположить, что

$$-K \le f_u(t,u) \le 0, \quad \tau \le \frac{2}{K}. \tag{36}$$

5.3. Методы Рунге-Кутта

Метод Эйлера прост, однако дает низкую точность. Порядок точности можно повысить путем усложнения разностной схемы. Весьма распространены на практике схемы Рунге-Кутта второго и четвертого порядков точности. Порядок точности схемы Рунге-Кутта 2-го порядка

$$\frac{y_{n+1} - y_n}{\tau} = (1-\sigma) f(t_n, y_n) + \sigma f\big(t_n + \alpha\tau,\ y_n + \alpha\tau f(t_n, y_n)\big) \tag{37}$$

зависит от параметров α, σ. Погрешность аппроксимации схемы (37) имеет вид

$$\psi_n = (1-\sigma) f(t_n, u_n) + \sigma f\big(t_n + \alpha\tau,\ u_n + \alpha\tau f(t_n, u_n)\big) - (u_{n+1} - u_n)/\tau. \tag{38}$$

Используя разложение по формуле Тейлора (35), получим:

$$\psi_n = \tau(\sigma\alpha - 1/2) u_n'' + O(\tau^2).$$

Отсюда видно, что схема (37) имеет второй порядок аппроксимации $\psi_n = O\left(\tau^2\right)$ при выполнении условия

$$\sigma\alpha = 1/2. \tag{39}$$

Рассмотрим частные случаи схемы Рунге-Кутта второго порядка аппроксимации:

1) $\sigma = 1, \quad \alpha = 1/2$,

$$\left(y_{n+1} - y_n\right)/\tau = f\left(t_n + \frac{\tau}{2}, y_n + \frac{\tau}{2} f\left(t_n, y_n\right)\right); \tag{40}$$

2) $\sigma = 1/2, \quad \alpha = 1$,

$$\left(y_{n+1} - y_n\right)/\tau = \frac{1}{2}\left[f\left(t_n, y_n\right) + f\left(t_n + \tau, y_n + \tau f\left(t_n, y_n\right)\right)\right]. \tag{41}$$

Приведем формулы для схемы Рунге-Кутта четвертого порядка точности:

$$\begin{gathered} \left(y_{n+1} - y_n\right)/\tau = \frac{1}{6}\left[K_1\left(y_n\right) + 2K_2\left(y_n\right) + 2K_3\left(y_n\right) + K_4\left(y_n\right)\right], \\ n = 0, 1, \ldots, N-1, \quad y_0 = u_0, \end{gathered} \tag{42}$$

где

$$\begin{gathered} K_1 = f\left(t_n, y_n\right), \quad K_2 = f\left(t_n + \tau/2, y_n + \tau K_1/2\right), \\ K_3 = f\left(t_n + \tau/2, y_n + \tau K_2/2\right), \quad K_4 = f\left(t_n + \tau, y_n + \tau K_3\right). \end{gathered} \tag{43}$$

При определении y_{n+1} по заданному y_n надо четыре раза вычислить правую часть.

Все методы Рунге-Кутта являются явными (y_{n+1} вычисляется по явной формуле) и одношаговыми (для определения y_{n+1} надо сделать один шаг на сетке от t_n к t_{n+1}). Явные схемы, как правило, являются условно устойчивыми, т.е. выбор шага интегрирования τ необходимо подчинять условию устойчивости. В случае схемы Эйлера τ выбирается из условия (36). Для схемы Рунге-Кутта второго порядка точности условие устойчивости схемы то же, что и для схемы Эйлера, т.е., $\tau \le 2/K$. На практике устойчивость схемы проверяют так: проводят вычисления на последовательности сеток с шагами $\tau_1 = \tau, \tau_2 = \tau_1/2$, добиваясь совпадения решения в соответствующих узлах сетки с требуемой точностью.

6. РАЗНОСТНЫЙ МЕТОД РЕШЕНИЯ КРАЕВЫХ ЗАДАЧ ДЛЯ ОБЫКНОВЕННЫХ ДИФФЕРЕНЦИАЛЬНЫХ УРАВНЕНИЙ ВТОРОГО ПОРЯДКА

6.1. Исходная задача

Рассмотрим первую краевую задачу для обыкновенного дифференциального уравнения второго порядка

$$Lu \equiv \left(k(x)u'\right)' - q(x)u = -f(x), \quad 0 < x < 1,$$
$$k(x) \geq c > 0, \; q(x) \geq 0, \quad u(0) = \mu_1, \quad u(1) = \mu_2. \tag{44}$$

Такое уравнение описывает стационарное, т.е. не меняющееся во времени распределение температуры. Задача (44) имеет единственное решение, если $k(x), q(x), f(x)$ – кусочно-непрерывные функции.

6.2. Трехточечные разностные схемы

Введем на отрезке [0,1] равномерную сетку $\overline{\omega}_h = \left\{x_i = ih, \, i = 0,1,\ldots,N, \, h = 1/N\right\}$ и выберем трехточечный шаблон $\left(x_{i-1}, x_i, x_{i+1}\right)$, на котором запишем разностную схему, аппроксимирующую задачу (44):

$$\Lambda y \equiv \frac{1}{h}\left[a_{i+1}\frac{y_{i+1} - y_i}{h} - a_i\frac{y_i - y_{i-1}}{h}\right] - d_i y_i = -\varphi_i,$$
$$i = 1,2,\ldots,N-1, \quad y_0 = \mu_1, \, y_N = \mu_2, \tag{45}$$

где $\quad a_i = k\left(x_i - \dfrac{h}{2}\right) = k_{i-1/2}, \qquad d_i = q(x_i), \qquad \varphi_i = f(x_i).$

При таком выборе $a_i, d_i,$ и φ_i схема (45) имеет второй порядок аппроксимации, т.е. невязка $\psi = \Lambda u + \varphi = O\left(h^2\right)$, если $k(x), \, q(x),$ и $f(x) \in C^{(2)}$ (имеют непрерывные производные до второго порядка), $u(x) \in C^{(4)}$. При этих предположениях схема (45) имеет второй порядок точности, т. е. $\|z\|_C = \|y - u\|_C = \max\limits_{0 \leq x_i \leq 1}|y_i - u_i| = O\left(h^2\right)$.

6.3. Решение разностных краевых задач методом прогонки

Разностная краевая задача (45) может быть записана в виде

$$A_i y_{i-1} - C_i y_i + B_i y_{i+1} = -F_i, \quad i = 1,2,\ldots,N-1, \tag{46}$$

$$y_0 = \chi_1 y_1 + \mu_1, \; y_N = \chi_2 y_{N-1} + \mu_2, \tag{47}$$

причем, $\chi_1 = \chi_2 = 0$ в случае первой краевой задачи.

Имеем систему линейных алгебраических уравнений с трехдиагональной матрицей размера $(N+1)\times(N+1)$. Для первой краевой задачи соответствующая матрица имеет размерность $(N-1)\times(N-1)$.

Для решения краевой задачи (46), (47) можно использовать следующий метод исключения, называемый методом прогонки. Предположим, что имеет место соотношение

$$y_i = \alpha_{i+1} y_{i+1} + \beta_{i+1} \tag{48}$$

с неопределенными коэффициентами $\alpha_{i+1}, \beta_{i+1}$, и подставим $y_{i-1} = \alpha_i y_i + \beta_i$ в (46):

$$(A_i \alpha_i - C_i) y_i + B_i y_{i+1} = -(F_i + A_i \beta_i).$$

Разрешая относительно y_i и сравнивая это тождество с (48), находим:

$$\alpha_{i+1} = \frac{B_i}{C_i - A_i \alpha_i}, \qquad i = 1, 2, \ldots, N-1, \tag{49}$$

$$\beta_{i+1} = \frac{A_i \beta_i + F_i}{C_i - A_i \alpha_i}, \qquad i = 1, 2, \ldots, N-1. \tag{50}$$

Используем краевое условие (47) при i=0 для определения α_1, β_1. Из формул (47) и (48) для i=0 находим

$$\alpha_1 = \chi_1, \ \beta_1 = \mu_1. \tag{51}$$

Зная α_1 и β_1 и переходя от i к $i+1$ в формулах (49), (50), определим α_i и β_i для всех $i = 2, 3, \ldots, N$. Вычисления по формуле (48) ведутся путем перехода от $i+1$ к i (т.е. зная y_{i+1}, находим y_i), и для начала этих вычислений надо задать y_N. Определим y_N из краевого условия $y_N = \chi_2 y_{N-1} + \mu_2$ и условия (48) при i=N-1: $y_{N-1} = \alpha_N y_N + \beta_N$. Отсюда находим

$$y_N = \frac{\mu_2 + \chi_2 \beta_N}{1 - \alpha_N \chi_2}. \tag{52}$$

Затем вычисляем $y_i = \alpha_{i+1} y_{i+1} + \beta_{i+1}, \ i = N-1, N-2, \ldots 1, 0.$

Устойчивость метода прогонки. Формулы прогонки можно применять, если знаменатели дробей в (49) и (52) не обращаются в нуль. Достаточным условием этого являются неравенства:

$$\begin{gathered} |C_i| \le |A_i| + |B_i|, \qquad i = 1, 2, \ldots, N-1, \\ |\chi_1| \le 1, \quad |\chi_2| \le 1, \quad |\chi_1| + |\chi_2| < 2. \end{gathered} \tag{53}$$

При выполнении условий (53) задача (46), (47) имеет единственное решение, определяемое по формулам (49) – (51).

СПИСОК РЕКОМЕНДУЕМОЙ ЛИТЕРАТУРЫ

1. *Бахвалов Н.С., Жидков Н.П., Кобельков Г.М.* Численные методы. М., 1987.
2. *Волков Е.А.* Численные методы. М., 1987.
3. *Петров А.В. и др.* Вычислительная техника в инженерных и экономических расчетах. М., 1984.

4. *Самарский А.А.* Введение в численные методы. М., 1982.
5. *Самарский А.А*. Теория разностных схем. М.,1977
6. *Самарский А.А., Гулин А.В.* Численные методы. М., 1989.