

МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ,
МОЛОДЕЖИ И СПОРТА УКРАИНЫ
Национальный аэрокосмический университет им. Н.Е. Жуковского
"Харьковский авиационный институт"

А.В. Попов, И.К. Васильева

**ВЫЧИСЛИТЕЛЬНЫЕ МЕТОДЫ В ПРОЕКТИРОВАНИИ
РАДИОЭЛЕКТРОННОЙ АППАРАТУРЫ**

Учебное пособие по лабораторному практикуму

Харьков «ХАИ» 2011

УДК 621.396.6.001.63 (076.5)

П58

Описано лабораторні роботи з дисципліни "Математичні методи і моделі в проектуванні РЕА", яка входить до програми підготовки бакалаврів за напрямом навчання "Радіоелектронні апарати". Викладено відомості з теорії обчислювальних методів, наведено постановку задач, методику їх числового розв'язання й приклади виконання робіт.

Лабораторні роботи орієнтовані на використання програми математичних розрахунків і моделювання Mathcad.

Для студентів факультету радіотехнічних систем літальних апаратів.

Рецензенты: д-р техн. наук, проф. В. А. Батаев;
д-р техн. наук, проф. Г. И. Хлопов

Попов, А. В.

П58 Вычислительные методы в проектировании радиоэлектронной аппаратуры [Текст]: учеб. пособие по лаб. практикуму / А. В. Попов, И. К. Васильева. – Х. : Нац. аэрокосм. ун-т им. Н. Е. Жуковского "Харьк. авиац. ин-т", 2011. – 90 с.

Описаны лабораторные работы по дисциплине "Математические методы и модели в проектировании РЭА", которая входит в программу подготовки бакалавров по направлению обучения "Радиоэлектронные аппараты". Изложены сведения из теории вычислительных методов, приведены постановка задач, методика их численного решения и примеры выполнения работ.

Лабораторные работы ориентированы на использование программы математических расчетов и моделирования Mathcad.

Для студентов факультета радиотехнических систем летательных аппаратов.

Ил. 14. Табл. 17. Библиогр.: 6 назв.

УДК 621.396.6.001.63 (076.5)

© Попов А. В., Васильева И. К., 2011

© Национальный аэрокосмический университет им. Н.Е. Жуковского "Харьковский авиационный институт", 2011

ВВЕДЕНИЕ

Методы вычислительной математики широко применяют при решении различных прикладных задач в науке, технике, производстве. ЭВМ позволяют точно и быстро выполнять стандартные математические операции, громоздкие и многократно повторяющиеся вычисления, графически представлять полученные результаты. Поэтому ЭВМ является важнейшей составной частью методики проектирования, применяемой в современной технике.

Основные типы инженерных задач, решаемые с помощью ЭВМ:

- нахождение корней алгебраических и трансцендентных уравнений;
- решение обыкновенных дифференциальных уравнений и дифференциальных уравнений в частных производных;
- задачи оптимизации;
- обработка массивов числовых данных.

В данной работе рассмотрены методы решения этих задач, за исключением оптимизационных.

Процесс решения задачи согласно выбранному вычислительному методу описывается в виде алгоритма. Чтобы алгоритм был реализован на ЭВМ, необходимо написать соответствующую программу. В настоящее время программирование в задачах технического характера походит на конструирование программ из готовых программных модулей. Такой подход к программированию целесообразен, так как экономит время, позволяет использовать навыки математиков и программистов, дает возможность сконцентрировать усилия на главной цели – решение научно-технической задачи. За многие годы накоплены обширные библиотеки научных подпрограмм, в первую очередь, на языке FORTRAN, предназначенных для решения типовых задач. Кроме того, имеются различные математические пакеты, реализующие численные методы, а также способные выполнять аналитические математические преобразования. Поэтому подготовка специалистов в области проектирования РЭА должна сочетать теорию вычислительных методов и их практическую реализацию на ЭВМ как путем написания собственных программ, так и с помощью средств современных математических пакетов. Такой подход позволит сформировать понимание математического содержания конкретного метода и умение использовать современные программные средства при решении конкретных задач проектирования.

Для каждого численного метода, рассмотренного в данной работе, приведен пример его программной реализации в пакете Mathcad.

Лабораторная работа № 1

ЧИСЛЕННОЕ РЕШЕНИЕ АЛГЕБРАИЧЕСКИХ И ТРАНСЦЕНДЕНТНЫХ УРАВНЕНИЙ

Цель работы:

- ознакомиться с численными методами решения нелинейных уравнений;
- вычислить с точностью $\varepsilon = 10^{-3}$ действительные корни заданных уравнений, используя различные методы: половинного деления, комбинированный, простой итерации;
- получить навыки реализации алгоритмов численных методов в пакете Mathcad.

Теоретические сведения

В процессе проектирования новых изделий или технологических процессов часто решают нелинейные уравнения; они могут быть самостоятельной задачей или составной частью более сложных задач. В обоих случаях практическая ценность численного метода в значительной мере определяется быстротой и эффективностью получения решения.

Нелинейные уравнения с одной переменной подразделяют на *алгебраические* и *трансцендентные*.

Уравнение

$$f(x) = 0 \quad (1.1)$$

называется алгебраическим, если функция $f(x)$ – алгебраическая, т.е. может быть представлена в виде суммы целых степеней x :

$$f(x) = a_0 + a_1x + \dots + a_{n-1}x^{n-1} + a_nx^n.$$

Алгебраическое уравнение порядка n имеет n корней, которые могут быть действительными или комплексными.

Если все коэффициенты a_i – действительные, то все комплексные корни образуют комплексно-сопряженные пары.

Число положительных действительных корней меньше или равно числу перемен знаков в последовательности коэффициентов a_i .

Число отрицательных действительных корней меньше или равно числу перемен знаков в последовательности коэффициентов a_i при замене x на $-x$.

Уравнение (1.1) называется трансцендентным, если оно содержит тригонометрические или другие специальные функции, например $\lg(x)$, e^x .

Решить уравнение (1.1) означает следующее:

- установить, имеет ли оно корни;
- определить число корней;
- найти значения корней с заданной точностью.

Отделение корней – процедура нахождения отрезков, на которых уравнение (1.1) имеет только один корень. Такие отрезки называют *интервалами локализации корня* или *интервалами неопределенности*.

В большинстве случаев корни можно отделить графически. Для этого достаточно построить график функции $f(x)$ и определить отрезки, на которых $f(x)$ имеет только одну точку пересечения с осью абсцисс. В сомнительных случаях графическое отделение корней необходимо подкреплять вычислениями. При этом используют следующие положения:

- если непрерывная функция $f(x)$ принимает на концах отрезка $[a, b]$ значения разных знаков (т.е. $f(a)f(b) < 0$), то уравнение (1.1) имеет на этом отрезке, по меньшей мере, один корень;
- если функция $f(x)$ строго монотонна ($f'(x)$ на отрезке $[a, b]$ не меняет знак), то корень – единственный.

Метод половинного деления. Пусть найден отрезок $[a_0, b_0]$, на котором находится единственный корень ξ уравнения (1.1). Вычислим среднее значение x на отрезке:

$$x_{\text{cp}} = \frac{a_0 + b_0}{2}.$$

Если $f(x_{\text{cp}}) = 0$, то $\xi = x_{\text{cp}}$ и задача решена.

Если $f(x_{\text{cp}}) \neq 0$, то выбираем ту из половин отрезка $[a_0, b_0]$, на концах которой функция $f(x)$ имеет разные знаки.

Для нового суженного отрезка $[a_1, b_1]$ повторяем те же действия и т.д. В результате на некотором шаге этой итерационной процедуры получаем или точный корень уравнения, или последовательность вложенных друг в друга отрезков $[a_0, b_0], [a_1, b_1], \dots, [a_n, b_n]$.

Доказано, что $\lim_{n \rightarrow \infty} a_n = \lim_{n \rightarrow \infty} b_n = \xi$.

Если необходимо вычислить корень уравнения с точностью до ε , то отрезок делим до тех пор, пока не выполнится условие

$$|b_n - a_n| \leq 2\varepsilon. \quad (1.2)$$

Приближенным значением корня считаем середину отрезка $[a_n, b_n]$:

$$\xi = \frac{a_n + b_n}{2}. \quad (1.3)$$

Функция, реализующая метод половинного деления:

$$\begin{array}{l}
 \mathbf{Half}(f, a, b, \varepsilon) := \left| \begin{array}{l}
 L \leftarrow |b - a| \\
 \mathbf{while} \ L > 2 \cdot \varepsilon \\
 \left| \begin{array}{l}
 c \leftarrow \frac{a + b}{2} \\
 \mathbf{break} \ \mathbf{if} \ |f(c)| < \varepsilon \\
 b \leftarrow c \ \mathbf{if} \ f(c) \cdot f(a) < 0 \\
 a \leftarrow c \ \mathbf{otherwise} \\
 L \leftarrow |b - a|
 \end{array} \right. \\
 c
 \end{array} \right.
 \end{array}$$

Функция, реализующая метод половинного деления и возвращающая значения приближения к корню уравнения на каждом шаге:

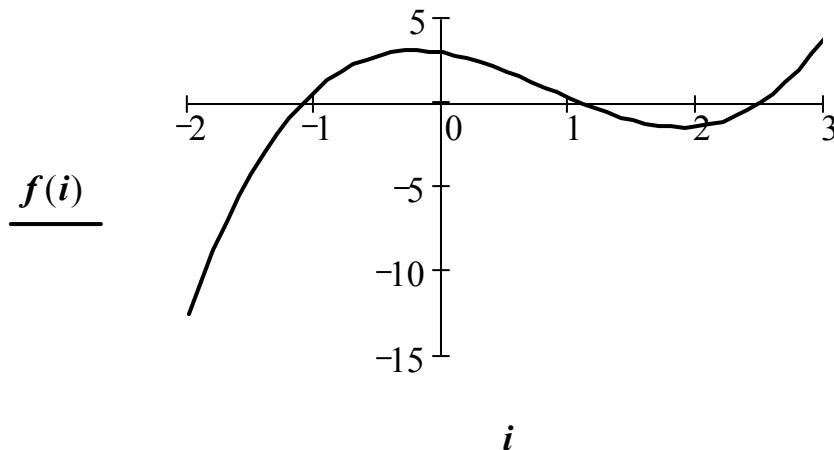
$$\begin{array}{l}
 \mathbf{Half}_i(f, a, b, \varepsilon) := \left| \begin{array}{l}
 L \leftarrow |b - a| \\
 i \leftarrow 0 \\
 \mathbf{while} \ L > 2 \cdot \varepsilon \\
 \left| \begin{array}{l}
 c \leftarrow \frac{a + b}{2} \\
 \mathbf{return} \ c \ \mathbf{if} \ |f(c)| < \varepsilon \\
 b \leftarrow c \ \mathbf{if} \ f(c) \cdot f(a) < 0 \\
 a \leftarrow c \ \mathbf{otherwise} \\
 L \leftarrow |b - a| \\
 X_{i,0} \leftarrow i \\
 X_{i,1} \leftarrow c \\
 i \leftarrow i + 1
 \end{array} \right. \\
 X
 \end{array} \right.
 \end{array}$$

Аргументы функций: f – имя функции в левой части уравнения; a , b – границы отрезка локализации корня; ε – точность вычисления.

Пример. Вычислить с точностью до $\varepsilon = 0,01$ действительные корни уравнения $x^3 - 2,5x^2 - 1,2x + 3 = 0$.

Алгебраическое уравнение третьей степени имеет три корня. Построим график функции

$$f(x) := x^3 - 2.5 \cdot x^2 - 1.2 \cdot x + 3 \quad i := -2, -1.9 \dots 3$$



Найдем три отрезка локализации корней уравнения:

$$[-2, -1], [1, 2], [2, 3].$$

Зададим точность вычислений

$$\varepsilon := 0.01$$

Определим с помощью функции $Half(f, a, b, \varepsilon)$ корни уравнения x_1, x_2, x_3 и значения функции $f(x)$ в каждом из них:

$$x1 := Half(f, -2, -1, \varepsilon) \quad x1 = -1.109 \quad f(x1) = -0.111$$

$$x2 := Half(f, 1, 2, \varepsilon) \quad x2 = 1.094 \quad f(x2) = 5.219 \times 10^{-3}$$

$$x3 := Half(f, 2, 3, \varepsilon) \quad x3 = 2.5 \quad f(x3) = 0$$

Получим значения приближений к корням x_1 и x_3 на каждом шаге:

$$Xi1 := Half_i(f, -2, -1, \varepsilon) \quad Xi1 = \begin{pmatrix} 0 & -1.5 \\ 1 & -1.25 \\ 2 & -1.125 \\ 3 & -1.063 \\ 4 & -1.094 \\ 5 & -1.109 \end{pmatrix}$$

$$Xi3 := Half_i(f, 2, 3, \varepsilon) \quad Xi3 = 2.5$$

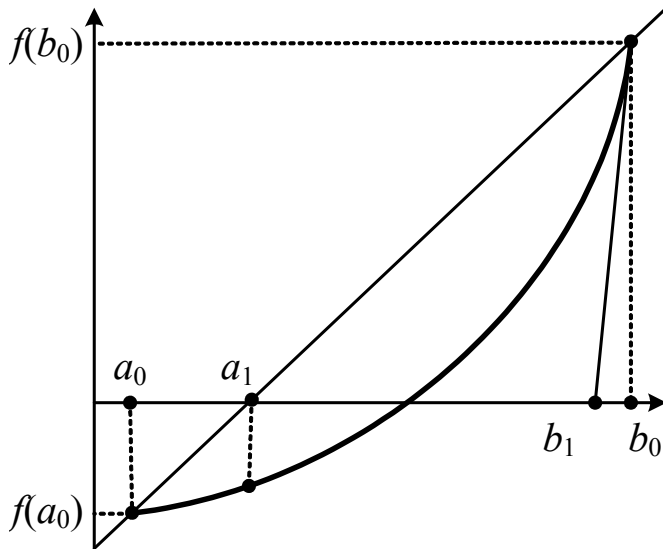


Рис. 1.1. Комбинированный метод

Комбинированный метод.

Дано уравнение $f(x) = 0$. Пусть найден отрезок $[a_0, b_0]$, на концах которого функция $y = f(x)$ имеет разные знаки, и производные $f'(x)$ и $f''(x)$ на этом отрезке сохраняют знак. Через b_0 обозначим тот конец отрезка, где знак функции $f(x)$ и знак ее второй производной $f''(x)$ совпадают. Таким образом, на рассматриваемом отрезке находится

единственный действительный корень уравнения ξ .

Комбинированный метод (рис. 1.1) заключается в поочередном применении метода хорд и метода касательных.

Через точки $(a_0, f(a_0))$, $(b_0, f(b_0))$ проводим хорду. Точку пересечения хорды с осью Ox обозначим через a_1 .

В точке $(b_0, f(b_0))$ проводим касательную к кривой $y = f(x)$. Точку пересечения касательной с осью Ox обозначим через b_1 .

Получим новый отрезок $[a_1, b_1]$, содержащий корень уравнения. Аналогично определим отрезки $[a_2, b_2], \dots, [a_n, b_n]$.

Расчетные формулы комбинированного метода имеют вид

$$a_{n+1} = a_n - \Delta a_n, \quad \Delta a_n = \frac{f(a_n)(b_n - a_n)}{f(b_n) - f(a_n)}, \quad (1.4)$$

$$b_{n+1} = b_n - \Delta b_n, \quad \Delta b_n = \frac{f(b_n)}{f'(b_n)}. \quad (1.5)$$

При заданной точности ε процесс вычисления прекращают, если выполняется условие (1.2), т.е. длина отрезка неопределенности $[a_n, b_n]$ становится достаточно малой ($\leq \varepsilon/2$) или значение функции $f(x)$ в середине интервала $[a_n, b_n]$ близко к нулю.

В качестве ответа принимают среднее арифметическое последних значений a_n и b_n , т.е.

$$\xi = \frac{a_n + b_n}{2}.$$

При отыскании корня комбинированный метод обеспечивает более быструю сходимость, чем метод половинного деления.

Подпрограмма-функция, реализующая комбинированный метод ($F1(x)$ и $F2(x)$ – вспомогательные функции):

$$F1(x) := \frac{d}{dx} f(x) \qquad F2(x) := \frac{d^2}{dx^2} f(x)$$

$$Comb(f, x1, x2, \varepsilon) := \left| \begin{array}{l} \text{if } F2(x1) \cdot f(x1) > 0 \\ \quad \left| \begin{array}{l} b \leftarrow x1 \\ a \leftarrow x2 \end{array} \right. \\ \text{otherwise} \\ \quad \left| \begin{array}{l} a \leftarrow x1 \\ b \leftarrow x2 \end{array} \right. \\ \text{while } |b - a| > 2 \cdot \varepsilon \\ \quad \left| \begin{array}{l} a \leftarrow a - \frac{f(a) \cdot (b - a)}{f(b) - f(a)} \\ b \leftarrow b - \frac{f(b)}{F1(b)} \end{array} \right. \\ c \leftarrow \frac{b + a}{2} \end{array} \right.$$

Пример. Вычислить с точностью до $\varepsilon = 0,001$ действительный корень уравнения $x^3 + 3x + 8 = 0$.

Рассмотрим функцию

$$f(x) := x^3 + 3 \cdot x + 8$$

На отрезке $[-2, -1]$ содержится корень данного уравнения, так как на концах этого отрезка функция имеет разные знаки:

$$f(-2) = -6 \qquad f(-1) = 4$$

Производная $f'(x) = 3x^2 + 3 > 0$ при всех x , поэтому уравнение имеет единственный действительный корень.

Вторая производная $f''(x) = 6x < 0$ при $-2 \leq x \leq -1$, поэтому через b_0 обозначаем конец отрезка $x = -2$, т.е. $b_0 = -2, a_0 = -1$.

Зададим точность вычислений

$$\varepsilon := 0.001$$

Найдем с помощью функции $Comb(f, a, b, \varepsilon)$ корень уравнения ξ и значение функции в корне $f(\xi)$:

$$\xi := Comb(f, -2, -1, \varepsilon)$$

$$\xi = -1.513 \quad f(\xi) = 9.115 \times 10^{-6}$$

Метод простой итерации. Для применения этого метода уравнение $f(x) = 0$ представим в виде

$$x = \varphi(x), \quad (1.6)$$

где $\varphi(x)$ – некоторая функция, непрерывная на отрезке $[a, b]$ (интервале локализации корня ξ исходного уравнения).

Выбираем произвольное $x_0 \in [a, b]$ и подставляем его в правую часть равенства (1.6). Получаем

$$\begin{aligned} x_1 &= \varphi(x_0), \\ x_2 &= \varphi(x_1), \\ x_n &= \overset{\dots}{\varphi}(x_{n-1}). \end{aligned} \quad (1.7)$$

Если последовательность x_0, x_1, \dots, x_n сходится, т.е. существует предел $\xi = \lim_{n \rightarrow \infty} x_n$, то $\lim_{n \rightarrow \infty} x_n = \varphi\left(\lim_{n \rightarrow \infty} x_{n-1}\right)$ или $\xi = \varphi(\xi)$.

Таким образом, ξ – корень уравнения $x = \varphi(x)$ и равнозначного ему уравнения $f(x) = 0$.

Для обеспечения сходимости итерационного процесса (1.7) (рис. 1.2) достаточно привести исходное уравнение к виду (1.6) так, чтобы выполнялось условие

$$|\varphi'(x)| < 1 \text{ при } a \leq x \leq b. \quad (1.8)$$

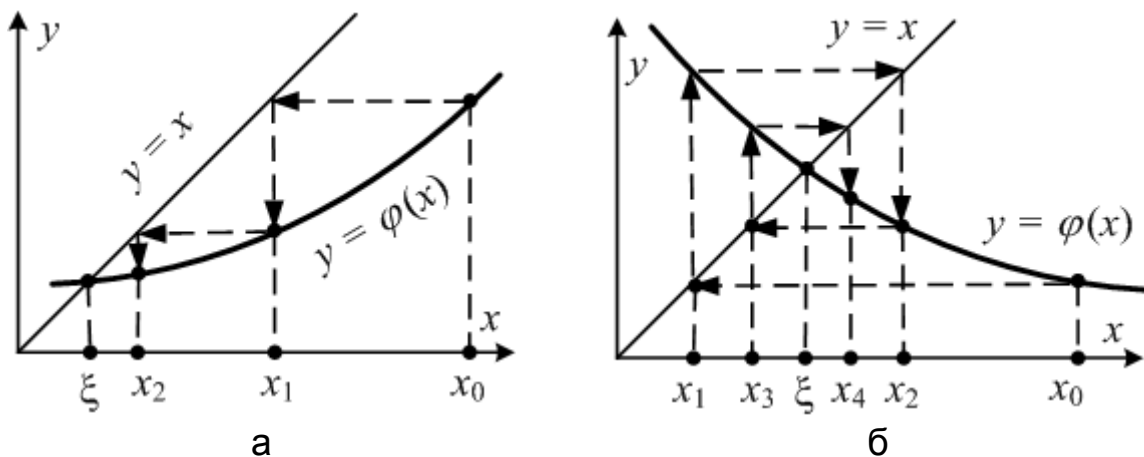


Рис. 1.2. Сходящийся итерационный процесс:
а – $0 < \varphi'(x) < 1$; б – $-1 < \varphi'(x) < 0$

Если условие сходимости (1.8) не выполняется, то итерационный процесс будет расходиться, как показано на рис. 1.3.

Обеспечить сходимость можно различными способами. Например, заменить $f(x) = 0$ равносильным уравнением $x = x - \lambda f(x)$. В этом случае $\varphi(x) = x - \lambda f(x)$.

Параметр λ подбираем так, чтобы $|\varphi'(x)| = |1 - \lambda f'(x)| < 1$ при $a \leq x \leq b$.

Если $0 < m \leq f'(x) \leq M$, где m – минимальное значение $f'(x)$ на отрезке $[a, b]$, M – максимальное, то можно выбрать $\lambda = 1/M$.

Отрезок $[a, b]$, на котором выполняется условие (1.8), называется *интервалом сходимости*.

Пример. Привести уравнение $x^3 + 4x - 1 = 0$ к виду, пригодному для применения метода итерации.

Единственный действительный корень данного уравнения находится на отрезке $[0, 1]$, так как $f(0) = -1$, $f(1) = 4$.

Приводим исходное уравнение к виду

$$x = \frac{1 - x^3}{4}. \quad (1.9)$$

В этом случае $\varphi(x) = (1 - x^3)/4$. Тогда $\varphi'(x) = -3x^2/4$. При $0 \leq x \leq 1$ $|\varphi'(x)| = 3x^2/4 \leq 3/4 < 1$. Таким образом, достаточное условие сходимости итерационного процесса (1.8) выполняется.

Уравнение (1.9) пригодно для применения метода итерации.

Пример. Привести уравнение $x^3 + 4x - 10 = 0$ к виду, пригодному для применения метода итерации.

Единственный действительный корень данного уравнения находится на отрезке $[1, 2]$. Рассмотренный в предыдущем примере способ приведения уравнения к итерационному виду в данном случае неприменим, так как при этом не удовлетворяется условие (1.8).

Заменяем исходное уравнение равносильным

$$x = x - \lambda(x^3 + 4x - 10).$$

В этом случае $\varphi(x) = x - \lambda(x^3 + 4x - 10)$, $\varphi'(x) = 1 - \lambda(3x^2 + 4)$.

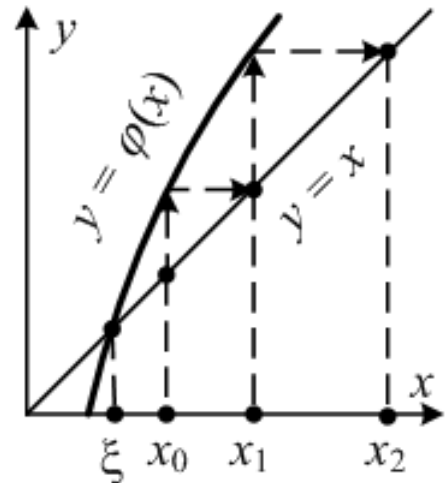


Рис. 1.3. Расходящийся процесс при $\varphi'(x) > 1$

Параметр λ находим из условия $|\varphi'(x)| < 1$ при $1 \leq x \leq 2$, т.е. $-1 < 1 - \lambda(3x^2 + 4) < 1$ или $0 < 1 - \lambda(3x^2 + 4) < 2$ при $1 \leq x \leq 2$.

Отсюда $0 < \lambda < 1/8$. Полагаем, например, $\lambda = 0,1$.

Исходное уравнение преобразуем к виду

$$x = x - 0,1(x^3 + 4x - 10), \quad (1.10)$$

причем

$$|\varphi'(x)| = |1 - 0,1(3x^2 + 4)| \leq 0,6 < 1 \text{ при } 1 \leq x \leq 2.$$

Уравнение (1.10) пригодно для применения метода итерации.

В качестве критерия остановки счета итерационного процесса можно использовать любое из следующих условий:

$$|f(x_n)| \leq \varepsilon, \quad (1.11)$$

$$|x_n - x_{n-1}| \leq \varepsilon, \quad (1.12)$$

где ε – заданная точность вычислений.

Подпрограмма-функция, реализующая метод простой итерации:

$$\text{Iter}(f, x_0, \varepsilon) := \left| \begin{array}{l} x_0 \leftarrow x_0 \\ i \leftarrow 0 \\ \text{while } |x_i - f(x_i)| > \varepsilon \\ \quad \left| \begin{array}{l} x_{i+1} \leftarrow f(x_i) \\ i \leftarrow i + 1 \end{array} \right. \\ x \end{array} \right.$$

Пример. Вычислить методом простой итерации с точностью до $\varepsilon = 0,0001$ действительный корень уравнения $x^3 + 4x - 10 = 0$.

Воспользуемся результатами, полученными в предыдущем примере. Корень уравнения находится на отрезке $[1, 2]$.

Исходное уравнение запишем в виде, пригодном для итерации (1.10):

$$\varphi(x) := x - 0,1 \cdot (x^3 + 4 \cdot x - 10)$$

Выбираем произвольное начальное приближение $x_0 \in [1, 2]$:

$$x_0 := 1$$

Получим массив значений последовательных приближений к корню уравнения

$$X := \text{Iter}(\varphi, x_0, 0.0001)$$

$$X = \begin{pmatrix} 1 \\ 1.5 \\ 1.563 \\ 1.556 \\ 1.557 \\ 1.557 \end{pmatrix}$$

Решением уравнения с точностью ε является последний элемент массива X , номер которого ni определим с помощью функции $last(X)$:

$$ni := last(X)$$

Получим корень уравнения X_{ni} и значение функции в корне $f(X_{ni})$:

$$X_{ni} = 1.557 \quad f(X_{ni}) = -1.349 \times 10^{-4}$$

Постановка задачи

Изучить численные методы нахождения корней нелинейных уравнений – половинного деления, комбинированный, метод простой итерации.

С помощью этих методов найти с точностью до $\varepsilon \leq 0,0001$ действительные корни уравнения (согласно заданному варианту).

Для процедуры отделения корней использовать графический метод (визуализация в пакете Mathcad).

Для процедуры поиска корня на интервале локализации (либо на интервале сходимости) создать в пакете Mathcad подпрограммы-функции, реализующие вычислительные схемы трех методов.

Сравнить эффективность указанных методов по критерию количества шагов (итераций) для достижения заданной точности ε .

Содержание отчета

Отчет о лабораторной работе должен содержать:

- постановку задачи;
- результат отделения действительного корня ξ уравнения;
- приведение заданного уравнения к виду, пригодному для применения метода итерации;
- последовательность приближений к корню x_0, x_1, \dots, x_n до достижения точности не хуже заданной;
- оценку погрешности вычислений Δx и представление полученных результатов в виде $\xi = x_n \pm \Delta x$;
- блок-схемы алгоритмов методов половинного деления, комбинированного и простой итерации;
- функции Mathcad, реализующие указанные алгоритмы.

Контрольные вопросы

1. Назовите исходные данные для приближенного решения нелинейных уравнений.
2. Зачем нужно отделять корни уравнения и какие существуют методы отделения корней?
3. Как отреагирует каждый из методов (половинного деления, комбинированный и простой итерации), если на исходном интервале $[a, b]$ будет два (и более) корня?
4. Насколько уменьшится исходный интервал $[a, b]$ после n итераций по методу половинного деления?
5. Как привести уравнение к итерационному виду?
6. Назовите достаточное условие сходимости простой итерации.
7. Почему в комбинированном методе проводить касательную к функции $y = f(x)$ нужно в той точке, где знаки f'_x и f''_{xx} совпадают?
8. Тождественны ли понятия «интервал неопределенности» и «интервал сходимости»? Если нет – то в чем различие?
9. От чего зависит число шагов методов половинного деления, комбинированного и простой итерации?
10. От чего зависят скорости сходимости указанных методов?
11. Перечислите критерии остановки счета.
12. Как оценить погрешность вычислений?

Варианты заданий

Номер варианта	f(x)	Номер варианта	f(x)
1	$x^3 - 2,8x^2 - 6,2x + 3,7$	11	$x^3 - 2,7x^2 - 3,5x + 0,8$
2	$x^3 - 9,9x^2 + 31,5x - 31,9$	12	$x^3 - 3,9x^2 + 4,4x - 1,4$
3	$x^3 + 0,3x^2 - 5,7x + 2,2$	13	$x^3 - 4,2x^2 + 3,1x + 2,7$
4	$x^3 - 0,8x^2 - 6,8x + 0,7$	14	$x^3 + 8,5x^2 + 21,8x + 15,6$
5	$x^3 - 0,9x^2 - 22,0x - 30,8$	15	$x^3 - 1,6x^2 - 2,4x + 0,3$
6	$x^3 - 5,4x^2 + 2,5x + 12,5$	16	$x^3 - 0,1x^2 - 4,6x + 2,2$
7	$x^3 - 4,7x^2 + 4,1x + 0,5$	17	$x^3 - 5,9x^2 + 11,1x - 6,7$
8	$x^3 - 7,5x^2 + 15,9x - 7,8$	18	$x^3 - 7,2x^2 + 16,5x - 11,9$
9	$x^3 - 4,4x^2 + 4,7x - 1,1$	19	$x^3 + 4,1x^2 + 3,6x - 0,4$
10	$x^3 - 4,8x^2 + 3,3x + 5,0$	20	$x^3 - 7,3x^2 + 13,5x - 5,0$

Лабораторная работа № 2

ИТЕРАЦИОННЫЕ МЕТОДЫ РЕШЕНИЯ СИСТЕМ АЛГЕБРАИЧЕСКИХ ЛИНЕЙНЫХ УРАВНЕНИЙ

Цель работы:

- ознакомиться с численными методами решения систем линейных уравнений (СЛУ);
- решить с точностью до $\varepsilon = 10^{-3}$ заданную систему уравнений, используя методы простой итерации и Зейделя;
- реализовать алгоритмы итерационных методов решения СЛУ в пакете Mathcad.

Теоретические сведения

Рассмотрим систему линейных алгебраических уравнений

$$\begin{cases} a_{11}x_1 + a_{12}x_2 + \dots + a_{1n}x_n = b_1; \\ a_{21}x_1 + a_{22}x_2 + \dots + a_{2n}x_n = b_2; \\ \vdots \\ a_{n1}x_1 + a_{n2}x_2 + \dots + a_{nn}x_n = b_n. \end{cases} \quad (2.1)$$

Обозначим

$$\mathbf{A} = \begin{bmatrix} a_{11} & a_{12} & \dots & a_{1n} \\ \vdots & & & \vdots \\ a_{n1} & a_{n2} & \dots & a_{nn} \end{bmatrix}, \quad \mathbf{X} = \begin{bmatrix} x_1 \\ \vdots \\ x_n \end{bmatrix}, \quad \mathbf{B} = \begin{bmatrix} b_1 \\ \vdots \\ b_n \end{bmatrix}.$$

Тогда СЛУ (2.1) можно записать в виде матричного уравнения

$$\mathbf{A} \mathbf{X} = \mathbf{B}. \quad (2.2)$$

Чтобы система (2.1) имела единственное решение, входящие в нее n уравнений должны быть линейно независимыми. Необходимое и достаточное условие этого

$$\det(\mathbf{A}) \neq 0.$$

Алгоритмы решения СЛУ бывают прямые и итерационные.

Прямые методы позволяют получить точное решение (с точностью до округления). Наиболее распространенные прямые методы основаны на приведении системы уравнений к треугольному виду: одно из уравнений содержит только одну неизвестную, а в каждом следующем – добавляется еще по одной неизвестной. Так, в **методе Гаусса** (методе исключения) сначала нормируют первое уравнение, разделив его коэффициенты на a_{11} , затем его последовательно ум-

ножают на первый коэффициент $a_{i,1}$ i -го уравнения ($i = 2, \dots, n$) и результат вычитают из i -го уравнения. В итоге первая переменная x_1 будет исключена из всех уравнений, кроме первого.

На следующем этапе эту же процедуру применяют к остальным $n - 1$ уравнениям. В результате из оставшихся $n - 2$ уравнений исключается вторая неизвестная x_2 . И так до тех пор, пока после n шагов вся система не будет приведена к треугольному виду

$$\begin{cases} \alpha_{11}x_1 + \alpha_{12}x_2 + \dots + \alpha_{1n}x_n = \beta_1; \\ \alpha_{22}x_2 + \dots + \alpha_{2n}x_n = \beta_2; \\ \vdots \\ \alpha_{nn}x_n = \beta_n. \end{cases}$$

В случае больших СЛУ прямые методы уступают итерационным. Итерационные схемы применяют к системам, предварительно приведенным к виду

$$\begin{cases} x_1 = \beta_1 + \alpha_{12}x_2 + \alpha_{13}x_3 + \dots + \alpha_{1n}x_n; \\ x_2 = \beta_2 + \alpha_{21}x_1 + \alpha_{23}x_3 + \dots + \alpha_{2n}x_n; \\ \vdots \\ x_n = \beta_n + \alpha_{n1}x_1 + \alpha_{n2}x_2 + \dots + \alpha_{n,n-1}x_{n-1}, \end{cases} \quad (2.3)$$

где $\beta_i = \frac{b_i}{a_{ii}}$, $a_{ii} \neq 0$; $\alpha_{ij} = \begin{cases} -\frac{a_{ij}}{a_{ii}}, & i \neq j; \\ 0, & i = j. \end{cases}$

В матричной форме

$$\mathbf{X} = \boldsymbol{\beta} + \boldsymbol{\alpha}\mathbf{X}. \quad (2.4)$$

В качестве нулевого приближения решения системы берут произвольный вектор $\mathbf{X}^{(0)}$. Целесообразно за компоненты начального вектора выбирать приближенные значения корней системы, которые находят грубой прикидкой. Также можно использовать столбец свободных членов, т.е. $\mathbf{X}^{(0)} = \boldsymbol{\beta}$.

Далее последовательно строим векторы:

$$\mathbf{X}^{(1)} = \boldsymbol{\beta} + \boldsymbol{\alpha}\mathbf{X}^{(0)} \text{ – первое приближение;}$$

$$\mathbf{X}^{(2)} = \boldsymbol{\beta} + \boldsymbol{\alpha}\mathbf{X}^{(1)} \text{ – второе приближение и т.д.}$$

Общий вид итерационной формулы для решения СЛУ

$$\mathbf{X}^{(k+1)} = \boldsymbol{\beta} + \boldsymbol{\alpha}\mathbf{X}^{(k)}, \quad k = 0, 1, 2, \dots \quad (2.5)$$

Таким образом, в **методе простой итерации** приближение

$\mathbf{X}^{(k+1)}$ вычисляется по предыдущему приближению $\mathbf{X}^{(k)}$ путем подстановки компонент $\mathbf{X}^{(k)}$ в правую часть всех уравнений системы (2.3):

$$x_i^{(k+1)} = \sum_{j=1}^n \alpha_{ij} x_j^{(k)} + \beta_i. \quad (2.6)$$

Если последовательность приближений $\mathbf{X}^{(k)}$ имеет предел при $k \rightarrow \infty$, то этот предел есть решение системы (2.1).

Для сходимости итерационного процесса достаточно, чтобы какая-либо норма (метрика) матрицы α была меньше единицы. Обычно используют одну из следующих метрик:

$$\rho_1(x, y) = \max_{1 \leq i \leq n} |x_i - y_i|, \quad (2.7)$$

$$\rho_2(x, y) = \sum_{i=1}^n |x_i - y_i|, \quad (2.8)$$

$$\rho_3(x, y) = \sqrt{\sum_{i=1}^n (x_i - y_i)^2}, \quad (2.9)$$

где $\mathbf{x} = (x_1, x_2, \dots, x_n)$ и $\mathbf{y} = (y_1, y_2, \dots, y_n)$ – точки n -мерного векторного пространства.

Для того, чтобы метод простой итерации для СЛУ (2.4) сходиллся, достаточно, чтобы выполнялось одно из условий:

– в пространстве с метрикой ρ_1 :

$$\alpha^* = \max_{1 \leq i \leq n} \sum_{j=1}^n |\alpha_{ij}| < 1 \quad (2.10)$$

(суммы по столбцам модулей элементов матрицы α должны быть меньше 1);

– в пространстве с метрикой ρ_2 :

$$\alpha^* = \max_{1 \leq j \leq n} \sum_{i=1}^n |\alpha_{ij}| < 1 \quad (2.11)$$

(суммы по строкам модулей элементов матрицы α должны быть меньше 1);

– в пространстве с метрикой ρ_3 :

$$\alpha^* = \sqrt{\sum_{i=1}^n \sum_{j=1}^n \alpha_{ij}^2} < 1 \quad (2.12)$$

(корень из суммы квадратов всех элементов матрицы α должен быть меньше 1).

Для установления момента прекращения итераций при достижении заданной точности можно использовать формулу

$$\rho(x^{(k-1)}, x^{(k)}) \leq \frac{\varepsilon(1 - \alpha^*)}{\alpha^*}. \quad (2.13)$$

Процесс итерации хорошо сходится (т.е. число приближений, необходимых для получения корней системы с заданной точностью, невелико), если элементы матрицы α малы по абсолютной величине ($|\alpha_{ij}| \ll 1$). Следовательно, модули диагональных коэффициентов a_{ii} системы (2.1) должны быть велики по сравнению с модулями недиагональных коэффициентов этой системы a_{ij} , $i \neq j$ (больше суммы модулей всех остальных коэффициентов, не считая свободных членов):

$$\begin{aligned} |a_{11}| &> |a_{12}| + |a_{13}| + \dots + |a_{1n}|; \\ |a_{22}| &> |a_{21}| + |a_{23}| + \dots + |a_{2n}|; \\ &\vdots \\ |a_{nn}| &> |a_{n1}| + |a_{n2}| + \dots + |a_{n,n-1}|. \end{aligned} \quad (2.14)$$

Для сходящегося итерационного процесса разности приближений $\Delta^{(k)} = X^{(k)} - X^{(k-1)} \rightarrow 0$ при $k \rightarrow \infty$.

При ручных вычислениях большое значение имеет свойство *самоисправляемости* сходящегося процесса, т.е. отдельная ошибка в вычислениях не отразится на окончательном результате, так как ошибочное приближение можно рассматривать как новый начальный вектор $X^{(0)}$.

Приведение СЛУ к виду, удобному для итерации. Пусть задана система линейных алгебраических уравнений (2.1). Система легко преобразуется к итерационному виду, если выполняется условие (2.14). Делим каждое уравнение исходной системы на диагональный элемент a_{ii} ($i = 1, \dots, n$) и находим x_i , т.е. разрешаем первое уравнение относительно x_1 , второе – относительно x_2 и т.д. Полученная таким образом система уравнений удовлетворяет достаточному условию сходимости (2.10).

Если диагональные элементы матрицы A не преобладают над остальными элементами, то из заданной системы выделим уравнения с коэффициентами, модули которых больше суммы модулей остальных коэффициентов уравнения. Каждое выделенное уравнение вписываем в такую строку новой системы, чтобы наибольший по модулю коэффициент стал диагональным. Из оставшихся неиспользованными и из выделенных уравнений системы составляем линейные

комбинации так, чтобы получить недостающие уравнения, причем диагональные коэффициенты по модулю должны быть больше суммы модулей всех остальных коэффициентов для каждого преобразованного уравнения. Необходимо, чтобы при составлении новой системы были использованы все уравнения исходной системы. Полученная таким образом система имеет матрицу коэффициентов A с преобладающими диагональными элементами и к ней применимы описанные выше преобразования.

Пример. Привести заданную систему к итерационному виду.

Дана система

$$\begin{aligned} (1) \quad & 1,36x_1 + 3,04x_2 + 3,37x_3 = 5,06; \\ (2) \quad & 2,35x_1 - 1,48x_2 + 1,02x_3 = 6,87; \\ (3) \quad & 4,26x_1 + 2,45x_2 - 1,96x_3 = -1,91. \end{aligned} \quad (2.15)$$

Система (2.15) не содержит уравнений с коэффициентами, модули которых больше суммы модулей остальных коэффициентов этих уравнений.

Путем элементарных преобразований приводим заданную систему к такому виду:

$$\begin{aligned} (2) + (3) \quad & 6,61x_1 + 0,97x_2 - 0,94x_3 = 4,96; \\ (2) - (1) \quad & 0,99x_1 - 4,52x_2 + 2,35x_3 = 1,81; \\ (3) - (1) \quad & 2,90x_1 + 0,59x_2 - 5,33x_3 = -6,97. \end{aligned} \quad (2.16)$$

Перед каждым уравнением системы (2.16) указано, каким образом получено данное уравнение из уравнений системы (2.15).

Например, первое уравнение системы (2.16) сформировано путем сложения второго и третьего уравнений системы (2.15).

При составлении новой системы (2.16) использованы все три уравнения системы (2.15).

Разрешим первое уравнение системы (2.16) относительно x_1 , второе – относительно x_2 , третье – относительно x_3 :

$$\begin{aligned} x_1 &= -0,1467x_2 + 0,1422x_3 + 0,7504; \\ x_2 &= 0,2190x_1 - 0,5199x_3 - 0,4004; \\ x_3 &= 0,5441x_1 - 0,1107x_2 + 1,3077. \end{aligned} \quad (2.17)$$

Система (2.17) пригодна для применения метода итерации.

Алгоритм решения системы методом итераций состоит из следующих шагов:

1. Привести систему (2.1) к виду с преобладающими диагональными коэффициентами.

2. Разделить каждое уравнение на соответствующий диагональный коэффициент.
3. Проверить выполнение условий сходимости (2.10) – (2.12).
4. Выбрать метрику ρ , для которой выполняется условие сходимости итерационного процесса.
5. Реализовать итерационный процесс (2.5), взяв за начальное приближение столбец свободных членов β .

Функции, возвращающие расстояния между точками в метрических пространствах ρ_1, ρ_2 , и ρ_3 :

$$\rho_1(x, y) := \begin{cases} z \leftarrow \overrightarrow{|x - y|} \\ \max(z) \end{cases}$$

$$\rho_2(x, y) := \begin{cases} z \leftarrow \overrightarrow{|x - y|} \\ \sum z \end{cases}$$

$$\rho_3(x, y) := \begin{cases} z \leftarrow x - y \\ |z| \end{cases}$$

Функция, реализующая метод простой итерации:

$$IterSLE(\alpha, \beta, \rho, c, \varepsilon) := \begin{cases} \delta \leftarrow \frac{\varepsilon \cdot (1 - c)}{c} \\ x_0 \leftarrow \beta \\ x_1 \leftarrow \alpha \cdot x_0 + \beta \\ \Delta \leftarrow \rho(x_0, x_1) \\ \mathbf{while} \ \Delta > \delta \\ \quad \begin{cases} x_0 \leftarrow x_1 \\ x_1 \leftarrow \alpha \cdot x_0 + \beta \\ \Delta \leftarrow \rho(x_0, x_1) \end{cases} \\ x_1 \end{cases}$$

Функция, реализующая метод простой итерации и возвращающая последовательность приближений к решению СЛУ:

$$\begin{array}{l}
 \text{XiterS}(\alpha, \beta, \rho, c, \varepsilon) := \left| \begin{array}{l}
 i \leftarrow 1 \\
 \delta \leftarrow \frac{\varepsilon \cdot (1 - c)}{c} \\
 x0 \leftarrow \beta \\
 X^{(0)} \leftarrow x0 \\
 x1 \leftarrow \alpha \cdot x0 + \beta \\
 X^{(1)} \leftarrow x1 \\
 \Delta \leftarrow \rho(x0, x1) \\
 \text{while } \Delta > \delta \\
 \quad \left| \begin{array}{l}
 x0 \leftarrow x1 \\
 x1 \leftarrow \alpha \cdot x0 + \beta \\
 \Delta \leftarrow \rho(x0, x1) \\
 X^{(i)} \leftarrow x1 \\
 i \leftarrow i + 1
 \end{array} \right. \\
 X
 \end{array} \right.
 \end{array}$$

В отличие от метода простой итерации в **методе Зейделя** на каждом шаге итерационного процесса при вычислении значения переменной $x_i^{(k)}$ учитывают уже найденные значения $x_1^{(k-1)}, \dots, x_{i-1}^{(k-1)}$. Это означает, что уточненное значение x_1 сразу же используют для вычисления x_2 . Затем по новым значениям x_1 и x_2 находят x_3 и т.д.

Области сходимости метода простой итерации и метода Зейделя не совпадают, а только пересекаются: существуют матрицы приведенных коэффициентов α , для которых метод Зейделя сходится, а метод простой итерации расходится, и наоборот.

Для сходимости метода Зейделя достаточно, чтобы выполнялось одно из условий – (2.10) или (2.11). В координатной записи итерационный процесс Зейделя имеет такой вид:

$$\begin{aligned}
x_1^{(k+1)} &= \beta_1 + \sum_{j=1}^n \alpha_{1j} x_j^{(k)}; \\
x_2^{(k+1)} &= \beta_2 + \alpha_{21} x_1^{(k+1)} + \sum_{j=2}^n \alpha_{2j} x_j^{(k)}; \\
&\vdots \\
x_n^{(k+1)} &= \beta_n + \sum_{j=1}^{n-1} \alpha_{nj} x_j^{(k+1)} + \alpha_{nn} x_n^{(k)}.
\end{aligned} \tag{2.18}$$

Функция, возвращающая массив приближений к решению СЛУ по методу Зейделя:

```

Zeidel( $\alpha$ ,  $\beta$ ,  $\rho$ ,  $\varepsilon$ ) :=
|  $n \leftarrow \text{rows}(\alpha) - 1$ 
|  $err \leftarrow 1$ 
|  $x0 \leftarrow \beta$ 
|  $x1 \leftarrow \beta$ 
|  $k \leftarrow 1$ 
|  $X^{(0)} \leftarrow x1$ 
| while  $err > \varepsilon$ 
|   | for  $i \in 0..n$ 
|   |   |  $tmp \leftarrow \sum_{j=0}^n \alpha_{i,j} \cdot x1_j + \beta_i$ 
|   |   |  $x1_i \leftarrow tmp$ 
|   |  $err \leftarrow \rho(x0, x1)$ 
|   |  $x0 \leftarrow x1$ 
|   |  $X^{(k)} \leftarrow x1$ 
|   |  $k \leftarrow k + 1$ 
|  $X$ 

```

Пример. Найти приближенное решение системы (2.15) с помощью методов простой итерации и Зейделя. Вычисления прекратить при достижении точности $\varepsilon = 0,01$.

Приведем исходную систему к итерационному виду (см. (2.17)).

Определим матрицу приведенных коэффициентов A и столбец приведенных свободных членов B .

$$A := \begin{pmatrix} 0 & -0.1467 & 0.1422 \\ 0.2190 & 0 & -0.5199 \\ 0.5441 & -0.1107 & 0 \end{pmatrix} \quad B := \begin{pmatrix} 0.7504 \\ -0.4004 \\ 1.3077 \end{pmatrix}$$

Вычислим норму матрицы A в пространстве с метрикой ρ_1 :
 $i := 0..2$

$$C_i := \left(\sum_{j=0}^2 |A_{i,j}| \right) \quad c := \max(C) \quad c = 0.739$$

Так как $c < 1$, то достаточное условие сходимости (2.10) методов простой итерации и Зейделя выполняется.

Решение системы с помощью метода простой итерации:

$$XI := \text{IterSLE}(A, B, \rho_1, c, 0.001)$$

$$XI = \begin{pmatrix} 1.231 \\ -1.23 \\ 2.114 \end{pmatrix}$$

Получим в матрице $XI = \{x_1^{(k)}, x_2^{(k)}, x_3^{(k)}\}$ значения приближений к решению заданной системы (символ T означает операцию транспонирования матрицы):

$$XI := \text{XiterS}(A, B, \rho_1, c, 0.01)^T \quad XI = \begin{pmatrix} 0.75 & -0.4 & 1.308 \\ 1.135 & -1.098 & 1.951 \\ 1.189 & -1.166 & 2.047 \\ 1.212 & -1.204 & 2.084 \\ 1.223 & -1.218 & 2.101 \\ 1.228 & -1.225 & 2.108 \\ 1.23 & -1.228 & 2.111 \end{pmatrix}$$

Чтобы получить решение системы (2.15) с точностью $\varepsilon = 0,01$ методом простой итерации потребовалось $k = 7$ шагов.

Решение системы с помощью метода Зейделя:

$$XZ := \text{Zeidel}(A, B, \rho I, 0.01)^T \quad XZ = \begin{pmatrix} 0.75 & -0.4 & 1.308 \\ 0.995 & -0.862 & 1.945 \\ 1.153 & -1.159 & 2.064 \\ 1.214 & -1.207 & 2.102 \\ 1.226 & -1.225 & 2.111 \\ 1.23 & -1.228 & 2.113 \end{pmatrix}$$

Заданная точность решения системы (2.15) по методу Зейделя достигнута за $k = 6$ шагов.

Получено решение системы:

$$x_1 = 1,23 \pm 0,01; x_2 = -1,23 \pm 0,01, x_3 = 2,11 \pm 0,01.$$

Постановка задачи

Изучить итерационные методы решения систем линейных алгебраических уравнений.

Определить, имеет ли единственное решение система уравнений с тремя неизвестными (согласно заданному варианту):

$$a_{11}x_1 + a_{12}x_2 + a_{13}x_3 = a_{14};$$

$$a_{21}x_1 + a_{22}x_2 + a_{23}x_3 = a_{24};$$

$$a_{31}x_1 + a_{32}x_2 + a_{33}x_3 = a_{34}.$$

Найти точное решение системы с помощью встроенной функции Mathcad $\text{Isolve}(A, B)$ (либо методом обратной матрицы).

Привести исходную систему к итерационному виду.

Проверить выполнение какого-либо из достаточных условий сходимости (2.10) – (2.12).

Используя методы простой итерации и Зейделя, найти корни заданной системы с точностью до $\varepsilon = 10^{-3}$.

Для процедуры поиска корней системы создать в пакете Mathcad подпрограммы-функции, реализующие вычислительные схемы итерационных методов решения СЛУ.

Сравнить эффективность исследуемых методов по критерию количества шагов для достижения заданной точности ε .

Содержание отчета

Отчет о лабораторной работе должен содержать:

- постановку задачи;
- схему проверки единственности решения заданной системы линейных уравнений;
- результаты поиска корней системы с помощью функции $lsolve(A, B)$ (либо методом обратной матрицы $X = A^{-1}B$);
- приведение системы к виду, пригодному для итерации;
- результаты приближенного решения системы методом простой итерации;
- результаты приближенного решения системы методом Зейделя;
- блок-схемы алгоритмов итерационных методов решения СЛУ;
- подпрограммы Mathcad, реализующие указанные алгоритмы.

Контрольные вопросы

1. Перечислите методы решения систем линейных уравнений.
2. В чем состоит метод Гаусса? Какое решение он позволяет получить – точное или приближенное?
3. Достоинства и недостатки метода Гаусса по сравнению с итерационными методами решения СЛУ.
4. Как привести систему уравнений к итерационному виду?
5. Что является достаточным условием сходимости процесса итерации?
6. Укажите исходные данные для приближенного решения систем линейных уравнений.
7. Как выбрать начальное приближение? Может ли выбор начального приближения повлиять на процесс итерации (скорость сходимости, число шагов, окончательный результат)?
8. От чего зависит число шагов итерационных методов?
9. От чего зависит скорость сходимости?
10. Как определить условие прекращения вычислений?
11. Сравните метод простой итерации и метод Зейделя – сходство и различия, достоинства и недостатки.
12. Предположим, что исходная система линейных уравнений недоопределена (переопределена). Можно ли такую систему решить каким-либо методом (Гаусса, простой итерации, Зейделя)? Если – нет, то почему, если – да, то будет ли полученное решение единственным?

Варианты заданий

Номер варианта	a_{11}	a_{12}	a_{13}	a_{14}	a_{21}	a_{22}	a_{23}	a_{24}	a_{31}	a_{32}	a_{33}	a_{34}
1	1,21	-4,05	2,11	4,25	0,75	1,21	-3,21	-7,42	2,27	5,66	-3,06	10,51
2	5,96	1,40	5,03	-7,41	2,99	1,25	-0,38	4,48	1,05	-3,57	1,92	-15,14
3	2,87	-2,67	-2,85	-2,14	3,55	0,71	-1,25	-0,95	1,13	-4,81	2,14	1,32
4	6,62	-2,65	-2,45	-2,57	5,21	-0,21	2,13	2,17	1,15	4,21	-1,75	-1,90
5	2,00	2,30	1,93	2,15	3,45	-0,58	1,21	1,55	1,25	4,21	-1,95	2,10
6	1,25	2,25	-3,75	2,00	1,75	-3,25	2,05	-1,80	2,35	1,25	1,85	6,70
7	0,75	1,23	-3,19	1,79	3,48	1,61	1,95	4,95	0,46	-5,26	5,32	14,71
8	3,58	2,77	2,34	-1,91	5,21	-2,13	0,49	3,42	1,17	3,90	-2,14	15,40
9	2,70	2,61	3,24	3,05	2,48	-0,18	1,71	2,55	1,20	3,48	-0,97	1,35
10	7,44	-2,46	2,74	-3,05	5,41	-1,25	2,01	2,57	1,15	3,81	-0,92	-1,15
11	1,26	4,20	-1,97	4,21	0,71	-1,91	3,88	-2,00	2,20	-4,79	3,15	-5,01
12	1,11	-4,83	2,15	-5,01	1,75	2,16	-5,01	2,25	2,43	5,52	-3,39	5,21
13	3,40	2,82	2,82	3,01	4,18	1,25	0,95	-1,15	1,71	-3,95	0,25	0,57
14	1,08	-3,50	1,90	4,15	3,01	-0,15	5,41	-1,27	0,06	-1,70	5,79	13,18
15	1,17	4,19	-1,77	4,25	1,39	-1,45	-4,60	-1,55	4,06	-5,42	3,88	-6,01
16	1,21	3,49	-0,99	3,75	1,50	-0,87	4,21	-1,05	1,28	-3,66	2,68	-3,91
17	1,10	3,80	-0,93	4,20	2,03	-1,21	4,75	-1,28	4,26	-5,06	2,93	-4,85
18	1,13	3,92	-2,17	14,21	2,41	-1,13	4,48	10,52	4,04	-6,03	2,63	-13,02
19	1,70	-3,94	0,26	1,85	0,78	-1,57	3,77	0,95	5,89	-2,70	-0,71	5,52
20	2,51	-0,20	1,75	2,15	1,52	-0,90	4,23	4,35	2,72	2,61	3,24	3,58
21	2,40	-1,15	4,50	0,09	6,38	1,77	-1,65	13,81	5,25	-2,15	0,50	11,41
22	1,41	-1,45	-4,61	1,58	2,56	2,77	6,33	2,85	5,18	-1,25	2,16	6,21
23	1,72	2,10	-4,95	2,01	3,51	0,75	-1,28	3,61	4,67	-4,10	0,89	4,55
24	5,39	-1,24	2,03	4,98	2,03	-1,24	-4,72	2,42	3,18	2,60	-5,67	3,52
25	3,44	-0,60	1,19	3,50	0,74	-1,90	3,90	0,85	5,45	1,72	3,14	5,05
26	0,80	-1,61	-3,76	-1,75	2,19	1,26	-0,94	1,29	0,93	-2,38	4,02	-2,50
27	-2,42	3,17	8,45	3,07	1,18	-2,14	-4,11	-1,38	4,46	6,45	8,17	3,95
28	1,24	-2,37	3,48	-3,17	0,76	4,21	-2,25	2,44	8,84	0,96	5,27	3,06
29	6,21	-4,52	5,16	5,24	-7,49	5,34	-4,16	-5,86	4,15	-9,48	9,20	1,71
30	4,42	12,64	-5,77	-1,56	0,16	-3,82	6,41	4,42	-7,14	4,98	-8,73	-2,67

Лабораторная работа № 3

МЕТОДЫ ОБРАБОТКИ ТАБЛИЧНЫХ ДАННЫХ. ИНТЕРПОЛЯЦИЯ

Цель работы:

- ознакомиться с методами интерполяции данных, представленных в виде таблиц;
- построить для функции, заданной таблично, интерполяционный полином Лагранжа;
- найти приближенные значения функции в промежуточных точках, используя полином Лагранжа и интерполяционные формулы Ньютона;
- реализовать алгоритмы полиномиальной интерполяции по методам Лагранжа и Ньютона в пакете Mathcad.

Теоретические сведения

Экспериментальные результаты обычно представляют в виде таблиц. При этом необходимо найти эмпирическую зависимость в виде некоторой функции $f(x)$, которая наилучшим образом описывала бы табличные данные. Если требуется, чтобы кривая $f(x)$ проходила через все заданные точки, – то это задача интерполяции.

Простейший вид интерполяции – линейная (кусочно-линейная) интерполяция, в основе которой лежит представление кривой на участке между точками (x_k, y_k) и (x_{k+1}, y_{k+1}) в виде прямой, проходящей через эти точки (рис. 3.1).

Уравнение такой прямой:

$$y = \frac{y_k(x - x_{k+1}) - y_{k+1}(x - x_k)}{x_k - x_{k+1}}. \quad (3.1)$$

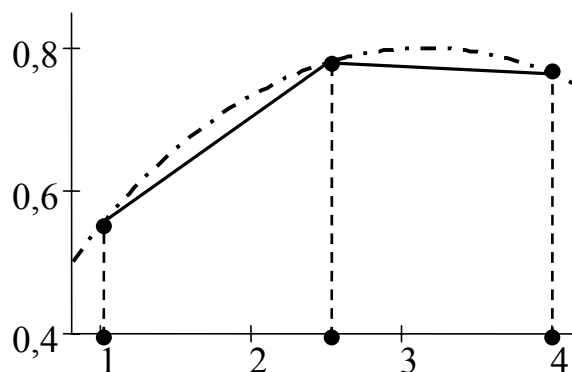


Рис. 3.1. Кусочно-линейная интерполяция

Таким образом, зная два табличных значения y_k и y_{k+1} , которые соответствуют координатам x_k и x_{k+1} , с помощью формулы (3.1) можно найти значение неизвестной функции $y = f(x)$ при любом x в интервале $[x_k, x_{k+1}]$.

Обычно полагают, что, используя большее число соседних точек и представляя истинную кривую более сложной линией, можно уточнить полученный результат. При этом ищут интерполирующую функцию в виде многочлена (полинома).

Математическая постановка задачи. Пусть на интервале $[a, b]$ заданы $n + 1$ точки x_0, x_1, \dots, x_n – узлы интерполяции и значения функции $f(x)$ в этих узлах y_0, y_1, \dots, y_n . Требуется построить интерполяционный многочлен $P_n(x)$ степени n , значения которого в узлах интерполяции совпадают с заданными значениями функции:

$$P_n(x_i) = f(x_i), \quad i = 0, \dots, n. \quad (3.2)$$

Доказано, что такой полином существует, притом единственный.

Интерполяционный полином $P_n(x)$ используют для приближенного вычисления значений неизвестной функции $f(x)$ для значений аргумента x , отличных от узлов x_i . Различают **интерполирование в узком смысле**, когда $x \in [x_0, x_n]$, т.е. значение x – промежуточное между x_0 и x_n , и **экстраполирование**, когда $x \notin [x_0, x_n]$.

Интерполяционный полином Лагранжа применяют для узлов интерполяции, расположенных с произвольным шагом.

Для построения полинома $P_n(x)$ рассмотрим вспомогательный многочлен $\omega_k(x)$, равный единице при $x = x_k$ и нулю в остальных узлах интерполяции:

$$\omega_k(x) = \begin{cases} 1 & \text{при } x = x_k, \\ 0 & \text{при } x = x_i \neq x_k. \end{cases} \quad (3.3)$$

Таким образом, многочлен $\omega_k(x)$ имеет вид

$$\omega_k(x) = \frac{(x - x_0)(x - x_1) \cdots (x - x_{k-1})(x - x_{k+1}) \cdots (x - x_n)}{(x_k - x_0)(x_k - x_1) \cdots (x_k - x_{k-1})(x_k - x_{k+1}) \cdots (x_k - x_n)}. \quad (3.4)$$

Вспомогательные полиномы второй степени $\omega_0(x)$, $\omega_1(x)$, $\omega_2(x)$, построенные по трем узлам интерполирования x_0, x_1, x_2 , показаны на рис. 3.2.

Интерполяционный многочлен можно записать так:

$$P_n(x) = \sum_{k=0}^n \omega_k(x) f(x_k). \quad (3.5)$$

$P_n(x)$ – полином степени не выше n , причем условие (3.2) выполняется:

$$P_n(x_i) = \sum_{k=0}^n \omega_k(x_i) y_k = \omega_i(x_i) y_i = y_i; \quad \forall i = 0, \dots, n.$$

Многочлен $P_n(x)$ (3.5) называется *интерполяционным многочленом Лагранжа*, а само равенство (3.5) – *формулой Лагранжа*.

Докажем единственность интерполяционного полинома.

Предположим противное. Пусть $\tilde{P}_n(x)$ – полином степени n , отличный от $P_n(x)$, и такой, что $\tilde{P}_n(x_i) = f(x_i) = y_i, \quad \forall i = 0, \dots, n$. Тогда полином $Q_n(x) = \tilde{P}_n(x) - P_n(x)$, степень которого, как очевидно, не выше n , обращается в ноль в $n + 1$ точках x_0, x_1, \dots, x_n . Следовательно, $Q_n(x) \equiv 0$ и $\tilde{P}_n(x) \equiv P_n(x)$.

Недостаток интерполирования данных по формуле (3.5) – полином Лагранжа практически непригоден для $n > 5$, так как при больших степенях полинома возникают большие всплески между узлами интерполяции (рис. 3.3).

Погрешность интерполяционной формулы Лагранжа

$$R_n(x) = f(x) - P_n(x)$$

удовлетворяет неравенству

$$|R_n(x)| \leq \frac{M_{n+1}}{(n+1)!} |\Pi_{n+1}(x)|,$$

где

$$M_{n+1} = \max_{a \leq x \leq b} |f^{(n+1)}(x)|;$$

$$\Pi_{n+1}(x) = (x - x_0)(x - x_1) \cdots (x - x_n).$$

Если интерполяционный полином находить не нужно, а требуется только вычислить его значения $P_n(x_j)$, $x_j \in [x_0, x_n]$ в промежуточных точках интервала интерполяции, то можно в

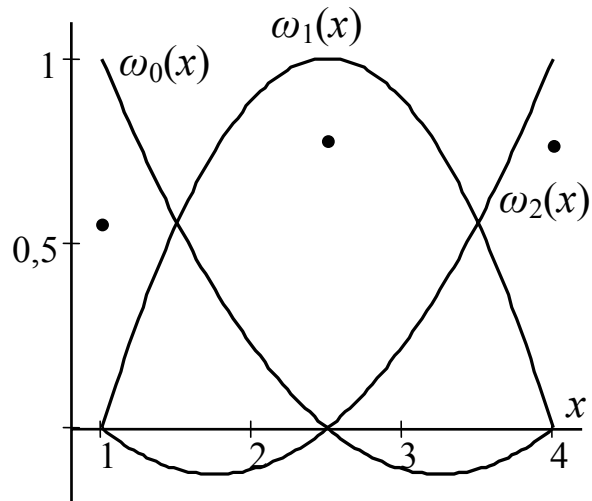


Рис. 3.2. Вспомогательные полиномы для узлов $x_0 = 1, x_1 = 2,5, x_2 = 4$

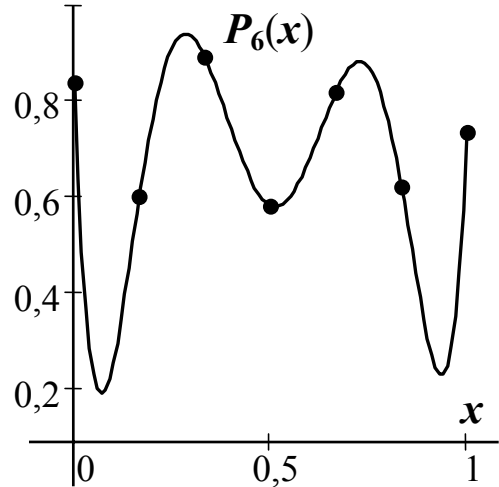


Рис. 3.3. Полином Лагранжа шестой степени

программе Mathcad определить вспомогательную функцию, реализующую произведение $\prod_{i=0, i \neq k}^n (x - x_i)$, и функцию, возвращающую значения интерполяционного полинома Лагранжа в точках x_j , как показано в примере.

Пример. Задана таблица данных.

x_i	-1	1	3	5	7
y_i	-3	2	3	0	5

Требуется по формуле Лагранжа найти значения y в промежуточных точках $x_5 = 2$, $x_6 = 4$, $x_7 = 6$. Построить график интерполяционного полинома Лагранжа, совместив его с табличными данными.

Зададим исходные данные

$$np := 5$$

$$i := 0 .. np - 1$$

$$y_i :=$$

Воспользуемся тем, что значения узлов интерполяции x_i заданы с равным шагом, и сформируем массив x в линейном цикле по i :

$$xmin := -1$$

$$xmax := 7$$

$$x_i := xmin + \frac{xmax - xmin}{np - 1} \cdot i$$

-3
2
3
0
5

Для построения графика полинома зададим 256 промежуточных точек на интервале интерполяции xx_{ii} :

$$N := 256 \quad ii := 0 .. N - 1 \quad xx_{ii} := xmin + \frac{xmax - xmin}{N - 1} \cdot ii$$

Определим вспомогательную функцию

$$W(a, k) := \prod_{i=0}^{last(x)} if[i = k, 1, (a - x_i)]$$

и функцию $Lagrang(z)$, вычисляющую значения полинома Лагранжа при заданных значениях аргумента z :

$$Lagrang(z) := \sum_{j=0}^{last(x)} y_j \cdot \frac{W(z, j)}{W(x_j, j)}$$

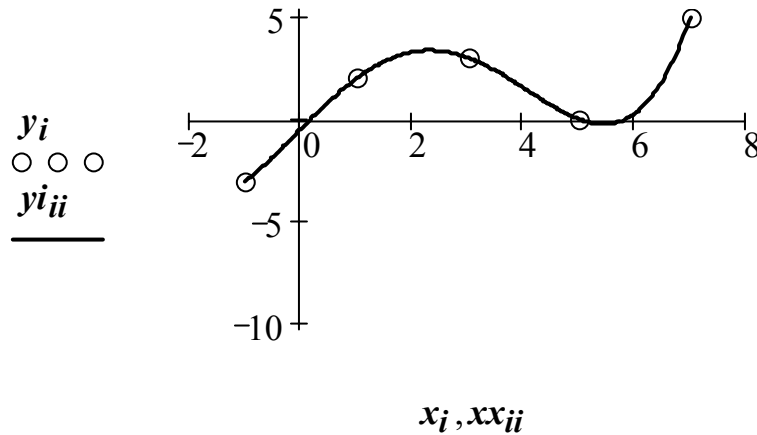
Найдем значения полинома Лагранжа при $x_5 = 2$, $x_6 = 4$, $x_7 = 6$:

$$Lagrang(2) = 3.281 \quad Lagrang(4) = 1.531 \quad Lagrang(6) = 0.281$$

Вычислим значения полинома Лагранжа в точках x_{ii} и сохраним полученные результаты в массиве y_i :

$$y_{ii} := \text{Lagrang}(x_{ii})$$

Построим график полинома Лагранжа $y_{ii} = P_4(x_{ii})$, совместив его с табличными данными $y_i = f(x_i)$.



Для больших таблиц данных, составленных при равноотстоящих значениях независимой переменной $x_i = x_0 + i h$ ($i = 0, \dots, n$, h – шаг), используют **разностные методы интерполяции**.

Наиболее распространен **метод Ньютона**.

Будем искать интерполяционный полином в виде

$$P_n(x) = a_0 + a_1(x - x_0) + a_2(x - x_0)(x - x_1) + \dots + a_n(x - x_0)\dots(x - x_{n-1}). \quad (3.6)$$

Коэффициенты a_i находят из уравнений $P_n(x_i) = y_i$, позволяющих записать следующую систему линейных уравнений:

$$\begin{cases} P_n(x_0) = a_0 = y_0; \\ P_n(x_1) = a_0 + a_1(x_1 - x_0) = y_1; \\ \vdots \\ P_n(x_n) = a_0 + \dots + a_n(x_n - x_0)(x_n - x_1)\dots(x_n - x_{n-1}) = y_n, \end{cases} \quad (3.7)$$

которая может быть решена с применением так называемых **конечных разностей**.

Для функции, заданной таблично с постоянным шагом h , **конечные разности первого порядка** определяют как разности между значениями функции в соседних узлах интерполяции:

$$\Delta y_i = y_{i+1} - y_i.$$

Из конечных разностей первого порядка можно образовать **конечные разности второго порядка**:

$$\Delta^2 y_i = \Delta y_{i+1} - \Delta y_i = y_{i+2} - 2y_{i+1} + y_i.$$

Аналогично получают выражения для конечных разностей третьего порядка:

$$\Delta^3 y_i = \Delta^2 y_{i+1} - \Delta^2 y_i = y_{i+3} - 3y_{i+2} + 3y_{i+1} - y_i.$$

Таким образом, конечные разности m -го порядка определяют через конечные разности $(m - 1)$ -го порядка:

$$\Delta^m y_i = \Delta^{m-1} y_{i+1} - \Delta^{m-1} y_i$$

или

$$\Delta^m y_i = y_{i+m} - my_{i+m-1} + \frac{m(m-1)}{2!} y_{i+m-2} - \dots + (-1)^m y_i. \quad (3.8)$$

Таблицу конечных разностей обычно записывают в следующем виде:

x	y	Δy	$\Delta^2 y$	$\Delta^3 y$	$\Delta^4 y$
x_0	y_0				
x_1	y_1	Δy_0			
x_2	y_2	Δy_1	$\Delta^2 y_0$	$\Delta^3 y_0$	
x_3	y_3	Δy_2	$\Delta^2 y_1$	$\Delta^3 y_1$	$\Delta^4 y_0$
x_4	y_4	Δy_3	$\Delta^2 y_2$		

Если табличные значения функции заданы с одинаковым числом десятичных знаков, то при оформлении таблицы конечных разностей $\Delta^m y_i$ записывают в единицах последнего разряда значений функции без нулей впереди.

С помощью конечных разностей коэффициенты полинома (3.6), являющиеся решением системы (3.7), можно записать так:

$$a_0 = y_0, \quad a_1 = \frac{\Delta y_0}{h}, \quad a_2 = \frac{\Delta^2 y_0}{2!h^2}, \dots, \quad a_n = \frac{\Delta^n y_0}{n!h^n}.$$

Подставив коэффициенты a_i в общий вид полинома Ньютона (3.6), получим **первую интерполяционную формулу Ньютона**

$$P_n(x) = y_0 + \frac{\Delta y_0}{h}(x - x_0) + \frac{\Delta^2 y_0}{2!h^2}(x - x_0)(x - x_1) + \dots \quad (3.9)$$

$$\dots + \frac{\Delta^n y_0}{n!h^n}(x - x_0)\dots(x - x_{n-1}).$$

Формулу (3.9) используют для интерполирования в точках x , близких к началу таблицы x_0 . При ручных вычислениях ее приводят к другому виду. Обозначим $\frac{x - x_0}{h} = q$, т.е. q – число шагов, на которые удалены x и x_0 .

Тогда

$$P_n(x) = y_0 + q\Delta y_0 + \frac{q(q-1)}{2!}\Delta^2 y_0 + \dots + \frac{q(q-1)\dots(q-n+1)}{n!}\Delta^n y_0.$$

Функция, вычисляющая конечные разности для массива, содержащего значения интерполируемой функции в узловых точках:

$$\Delta n f(y) := \left| \begin{array}{l} n \leftarrow \text{last}(y) \\ \text{for } i \in 0..n \\ \quad \Delta_{i,0} \leftarrow y_i \\ \text{for } k \in 1..n \\ \quad \text{for } i \in 0..n-k \\ \quad \quad \Delta_{i,k} \leftarrow \Delta_{i+1,k-1} - \Delta_{i,k-1} \\ \Delta \end{array} \right.$$

Конечные разности, входящие в формулу (3.9), расположены в верхней кривой строке таблицы конечных разностей.

При построении интерполяционного многочлена обычно учитывают не все заданные узлы x_i , $i = 0, \dots, n$, а их часть, что приводит к уменьшению степени полинома (3.9). Метод позволяет уточнять результат, используя дополнительные разности, причем нет необходимости начинать вычисления сначала. Поэтому, если неизвестно, сколько членов потребуется для построения интерполяционного полинома $P_m(x)$, $m < n$, их число можно увеличивать до тех пор, пока их вклад не станет пренебрежимо мал (т.е. пока конечные разности в соседних столбцах $\Delta^m y_0$ и $\Delta^{m+1} y_0$ не будут иметь одинаковые разряды). Тогда первая интерполяционная формула Ньютона примет вид

$$P_m(x) = P_m(x_0 + qh) = y_0 + q\Delta y_0 + \frac{q(q-1)}{2!}\Delta^2 y_0 + \dots + \frac{q(q-1)\dots(q-m+1)}{m!}\Delta^m y_0. \quad (3.10)$$

Пример. Задана таблица данных (значения функции $y = \sin(x)$).

$x_i, ^\circ$	10	20	30	40	50	60
y_i	0,17365	0,34202	0,50000	0,64279	0,76604	0,86603

Требуется по формуле Ньютона найти y при $x = 23^\circ$.

Зададим исходные данные в пакете Mathcad:

$$i := 0..5$$

$$h := 10$$

$$x_i := 10 + h \cdot i$$

Определим функцию $\Delta n f(y)$ и с ее помощью вычислим конечные разности (до пятого порядка включительно), значения которых сохраним в матрице Δy :

$$\Delta y := \Delta n f(y)$$

При выводе на экран умножим элементы матрицы на 10^5 , чтобы видеть величины в разряде сотысячных без нулей впереди.

$$y_i :=$$

0.17365
0.34202
0.50000
0.64279
0.76604
0.86603

$$\Delta y \cdot 10^5 =$$

17365	16837	-1039	-480	45	18
34202	15798	-1519	-435	63	0
50000	14279	-1954	-372	0	0
64279	12325	-2326	0	0	0
76604	9999	0	0	0	0
86603	0	0	0	0	0

Получим таблицу конечных разностей в виде треугольной матрицы 6×6 , где первый столбец содержит значения интерполируемой функции $\Delta y_{i,0} = y_i$, второй – значения конечных разностей первого порядка $\Delta y_{i,1} = \Delta y_i$, третий – значения конечных разностей второго порядка $\Delta y_{i,2} = \Delta^2 y_i$ и т.д.

Поскольку значение $x = 23^\circ$ находится между узлами $x_1 = 20^\circ$ и $x_2 = 30^\circ$, то за точку привязки x_0 возьмем x_1 . Тогда в первую интерполяционную формулу Ньютона (3.9) вместо x_0 подставим x_1 , вместо $y_0 - y_1$, а вместо конечных разностей k -го порядка $\Delta^k y_0$, расположенных в первой строке таблицы, – разности $\Delta^k y_1$, которые находятся во второй строке (им соответствуют элементы массива $\Delta y_{1,k}$).

Сначала используем только первую разность

$$P1(z) := y_1 + \frac{\Delta y_{1,1}}{h} \cdot (z - x_1)$$

Получим значение полинома первой степени при $x = 23^\circ$

$$P1(23) = 0.38941$$

Введем дополнительно первую разность

$$P2(z) := P1(z) + \frac{\Delta y_{1,2}}{2! \cdot h^2} \cdot (z - x_1) \cdot (z - x_2)$$

Найдем соответствующее значение полинома второй степени

$$P2(23) = 0.39101$$

Уточним результат с помощью третьей разности

$$P3(z) := P2(z) + \frac{\Delta y_{1,3}}{3! \cdot h^3} \cdot (z - x_1) \cdot (z - x_2) \cdot (z - x_3)$$

Получим значение $y(23)$ с использованием полинома третьей степени

$$P3(23) = 0.39075$$

Наконец, добавим к сумме четвертую разность

$$P4(z) := P3(z) + \frac{\Delta y_{1,4}}{4! \cdot h^4} \cdot (z - x_1) \cdot (z - x_2) \cdot (z - x_3) \cdot (z - x_4)$$

Окончательный результат – значение полинома четвертой степени при $x = 23^\circ$:

$$P4(23) = 0.39072$$

Конечную разность пятого порядка можно не учитывать, так как величины $\Delta^4 y_i$ и $\Delta^5 y_i$ имеют одинаковый разряд.

Действительно, полученное значение $y(23)$ близко к точному:

$$\sin(23 \cdot \text{deg}) = 0.39073$$

Из приведенного примера видно, как метод интерполяции Ньютона позволяет уточнять результат с помощью дополнительных конечных разностей.

Погрешность первой интерполяционной формулы Ньютона можно оценить по формуле

$$R_m(x) = \frac{q(q-1)\dots(q-m)}{(m+1)!} h^{m+1} f^{(m+1)}(\xi),$$

где ξ – некоторая точка интервала, содержащего узлы интерполяции.

Функция пакета Mathcad, реализующая вычисления по первой интерполяционной формуле Ньютона:

$$\begin{array}{l}
 \mathbf{Newton1}(x, y, xx) := \left| \begin{array}{l}
 n \leftarrow \text{last}(x) \\
 s \leftarrow y_0 \\
 \Delta \leftarrow \Delta n f(y) \\
 h \leftarrow x_1 - x_0 \\
 \text{for } k \in 1 .. n \\
 \quad \left| \begin{array}{l}
 p \leftarrow 1 \\
 a \leftarrow 1 \\
 \text{for } i \in 0 .. k - 1 \\
 \quad \left| \begin{array}{l}
 p \leftarrow p \cdot (xx - x_i) \\
 a \leftarrow a \cdot h \cdot (i + 1)
 \end{array} \right. \\
 s \leftarrow s + \Delta_{0,k} \cdot \frac{p}{a}
 \end{array} \right. \\
 s
 \end{array} \right.
 \end{array}$$

Аргументы функции *Newton1*: x – массив, содержащий координаты узловых точек; y – массив значений интерполируемой функции в узловых точках; xx – координата промежуточной точки интервала интерполирования. Данная функция возвращает значение интерполяционного полинома (3.9) в точке xx , $P_n(xx)$, при этом степень полинома n на единицу меньше числа узлов интерполяции x_i , $i = 0, \dots, n$. Если количество узлов велико, то интерполяцию проводят только по части данных, т.е. строят интерполяционный полином $P_m(x)$ (3.10) степени m , $m < n$.

Значение m степени интерполяционного полинома $P_m(x)$ можно определить по таблице конечных разностей, как указано выше. Тогда в качестве аргументов x и y функции *Newton1* следует указывать имена массивов длиной $m + 1$, содержащих координаты узловых точек и значения интерполируемой функции в этих точках.

Например, при описании входных параметров функции *Newton1* можно поступить следующим образом.

Определим переменную цикла

$$j := 0 .. m$$

Зная шаг h между узлами, найдем номер N элемента массива x , ближайшего (сверху) к значению промежуточной точки xx , $x_N < xx$:

$$N := \text{floor}\left(\frac{xx}{h}\right)$$

Сформируем массив u , содержащий $m + 1$ элемент массива x , причем $u_0 = x_N$:

$$u_j := x_{N+j}$$

Аналогичным образом из массива y получим массив z , содержащий $m + 1$ значение интерполируемой функции в точках u_j :

$$z_j := y_{N+j}$$

Тогда формат вызова функции *Newton1* примет вид

$$\text{Newton1}(u, z, xx)$$

(переменная xx должна быть определена выше).

Если точка интерполирования x лежит вблизи конечной точки таблицы x_n , то применяют **вторую интерполяционную формулу Ньютона**, которая получается, если искать интерполяционный полином в виде

$$P_n(x) = a_0 + a_1(x - x_n) + a_2(x - x_n)(x - x_{n-1}) + \dots + a_n(x - x_n)\dots(x - x_1). \quad (3.11)$$

Коэффициенты полинома (3.11) находят из условия (3.2) совпадения значений функции и интерполяционного полинома в узлах:

$$a_k = \frac{\Delta^k y_{n-k}}{k! h^k}. \quad (3.12)$$

Подставив значения коэффициентов (3.12) в уравнение (3.11) и обозначив $q = \frac{x - x_n}{h}$, получим вторую интерполяционную формулу Ньютона

$$P_n(x) = P_n(x_n + qh) = y_n + q\Delta y_{n-1} + \frac{q(q+1)}{2!}\Delta^2 y_{n-2} + \dots + \frac{q(q+1)\dots(q+n-1)}{n!}\Delta^n y_0. \quad (3.13)$$

Формула (3.13) содержит конечные разности, расположенные в нижней косой строке таблицы конечных разностей.

Для построения интерполяционного многочлена $P_m(x)$, $m < n$ вторую интерполяционную формулу Ньютона запишем так:

$$P_m(x) = y_n + q\Delta y_{n-1} + \frac{q(q+1)}{2!}\Delta^2 y_{n-2} + \dots \quad (3.14)$$

$$\dots + \frac{q(q+1)\dots(q+m-1)}{m!}\Delta^m y_{n-m}.$$

Погрешность формулы (3.14) имеет вид

$$R_m(x) = \frac{q(q+1)\dots(q+m)}{(m+1)!}h^{m+1}f^{(m+1)}(\xi),$$

где ξ – некоторая точка интервала, содержащего узлы интерполяции.

Функция, реализующая вычисления по второй интерполяционной формуле Ньютона:

```

Newton2(x, y, xx) :=
  n ← last(x)
  s ← y_n
  Δ ← Δnf(y)
  h ← x_1 - x_0
  for k ∈ 1..n
    p ← 1
    a ← 1
    for i ∈ 0..k-1
      p ← p · (xx - x_{n-i})
      a ← a · h · (i + 1)
    s ← s + Δ_{n-k, k} · p / a
  s
  
```

Аргументы функции *Newton2* соответствуют аргументам функции *Newton1*. Если заданы массивы x и y , содержащие по $n + 1$ элементов (координат узлов, расположенных с шагом h , и значений функции в узлах), то для вычисления значения полинома $P_m(x)$ (3.14) степени m ($m < n$) в точке xx следует определить массивы u и z :

$$N := \text{floor}\left(\frac{xx}{h}\right) + 1 \quad u_j := x_{N-m+j} \quad z_j := y_{N-m+j}$$

Таким образом, обращение к функции *Newton2* примет вид

$$\text{Newton2}(u, z, xx)$$

(переменная *xx* должна быть определена выше).

Пример. Функция задана таблично.

x_i	1,5	1,6	1,7	1,8	1,9	2,0	2,1	2,2
y_i	0,17609	0,20412	0,23045	0,25527	0,27875	0,30103	0,32222	0,34242

Вычислить приближенно значения функции *y* в точках $xx_1 = 1,53$; $xx_2 = 1,55$; $xx_3 = 1,57$; $xx_4 = 2,13$; $xx_5 = 2,15$; $xx_6 = 2,17$.

Зададим исходные данные в пакете Mathcad:

$$i := 0 .. 7$$

$$h := 0.1$$

$$x_i := 1.5 + h \cdot i$$

Точки xx_1 , xx_2 и xx_3 расположены в начале таблицы, поэтому воспользуемся первой интерполяционной формулой Ньютона:

$$\text{Newton1}(x, y, 1.53) = 0.18469$$

$$\text{Newton1}(x, y, 1.55) = 0.19033$$

$$\text{Newton1}(x, y, 1.57) = 0.1959$$

Точки xx_4 , xx_5 и xx_6 расположены в конце таблицы, поэтому применим вторую интерполяционную формулу Ньютона:

$$\text{Newton2}(x, y, 2.13) = 0.32838$$

$$\text{Newton2}(x, y, 2.15) = 0.33244$$

$$\text{Newton2}(x, y, 2.17) = 0.33646$$

$$y_i :=$$

0.17609
0.20412
0.23045
0.25527
0.27875
0.30103
0.32222
0.34242

Постановка задачи

Изучить методы интерполяции числовых данных.

Для функции, заданной таблично (согласно заданному варианту), по точкам x_0, x_1, x_2, x_3, x_4 получить интерполяционный полином Лагранжа $P_4(x)$. Найти приближенные значения функции в точках $\frac{x_i + x_{i+1}}{2}$, где $i = 0, \dots, 3$.

По 100 промежуточным точкам построить график интерполяционного полинома Лагранжа, совместив его с табличными данными.

Составить таблицу конечных разностей заданной функции, с помощью которой определить достаточную степень m интерполяционного полинома Ньютона $P_m(x)$.

По интерполяционным формулам Ньютона вычислить приближенные значения функции в точках $\frac{x_0 + x_1}{2}$ и $\frac{x_8 + x_9}{2}$.

Сравнить результаты интерполяции по методам Лагранжа и Ньютона.

Содержание отчета

Отчет о лабораторной работе должен содержать:

- постановку задачи;
- основные этапы построения интерполяционного полинома;
- проверку условий $P_m(x_i) = y_i, i = 0, \dots, m$;
- график интерполяционного полинома Лагранжа, совмещенный с заданными точками;
- таблицу конечных разностей заданной функции;
- приближенные значения функции в заданных промежуточных точках, найденные с помощью методов интерполяции Лагранжа и Ньютона;
- подпрограммы Mathcad, реализующие вычислительные схемы интерполяционных формул Лагранжа и Ньютона.

Контрольные вопросы

1. В чем заключается задача интерполяции?
2. Что такое экстраполяция? Можно ли с помощью формул Лагранжа и Ньютона ее осуществить?
3. Как определяется степень интерполяционного многочлена? Можно ли построить интерполяционный полином второй степени по двум, трем, четырем и т.д. узлам?
4. Что называется конечной разностью k -го порядка? Запишите выражение для разности порядка k непосредственно через табличные значения функции.
5. Чем определяется необходимое количество конечных разностей в формулах Ньютона?
6. От чего зависит погрешность интерполяции?
7. Сравните методы интерполяции Лагранжа и Ньютона: сходство и различия, достоинства и недостатки.

Варианты заданий

Номер варианта	1	2	3	4	5	6	7	8	9	10
Функция	А	Б	В	Г	Д	Е	Ж	З	И	К
x_0	0,0	0,2	0,4	0,6	0,8	1,0	1,1	1,2	1,3	1,4
x_9	0,9	1,1	1,3	1,5	1,7	1,9	2,0	2,1	2,2	2,3

Номер варианта	11	12	13	14	15	16	17	18	19	20
Функция	Л	М	Н	О	П	Р	С	Т	А	Б
x_0	0,1	0,3	0,5	0,7	0,9	1,5	1,6	1,7	2,0	2,4
x_9	1,0	1,2	1,4	1,6	1,8	2,4	2,5	2,6	2,9	3,3

А	Б	В	Г	Д	Е	Ж	З	И
0,94608	0,96285	0,97950	0,99602	1,01242	1,00000	0,99500	0,87758	0,47943
1,02868	1,04482	1,06083	1,07670	1,09244	0,99500	0,98007	0,82534	0,56464
1,10805	1,12351	1,13884	1,15402	1,16906	0,98007	0,95534	0,76484	0,64422
1,23961	1,19871	1,21331	1,22777	1,24207	0,95534	0,92106	0,69671	0,71736
1,25623	1,27023	1,28408	1,29776	1,31130	0,92106	0,87758	0,62161	0,78333
1,32468	1,33790	1,35096	1,36386	1,37660	0,87758	0,82534	0,54030	0,84147
1,38918	1,40159	1,41384	1,42592	1,43784	0,82534	0,76484	0,45360	0,89121
1,44959	1,46117	1,47260	1,48384	1,49491	0,76484	0,69671	0,36236	0,93204
1,50582	1,51655	1,52711	1,53751	1,54772	0,69671	0,62161	0,26750	0,96356
1,55777	1,56761	1,57735	1,58688	1,59623	0,62161	0,54030	0,16997	0,98545
К	Л	М	Н	О	П	Р	С	Т
0,38942	1,49156	1,50000	1,29608	0,62963	0,50000	1,00000	1,21823	1,87590
0,47943	1,46639	1,49156	1,21091	0,75708	0,62963	1,10978	1,32404	1,81309
0,56464	1,42491	1,46639	1,11375	0,88019	0,75708	1,21823	1,42594	1,71388
0,64422	1,36782	1,42491	1,00622	0,99688	0,88019	1,32404	1,52269	1,58222
0,71736	1,29608	1,36782	0,89015	1,10519	0,99688	1,42594	1,61312	1,42337
0,78333	1,21091	1,29608	0,76750	1,20328	1,10519	1,52269	1,69614	1,24365
0,84147	1,11375	1,21091	0,64033	1,28950	1,20328	1,61312	1,77074	1,05023
0,89121	1,00622	1,11375	0,51080	1,36240	1,28950	1,69614	1,83603	0,85082
0,93204	0,89015	1,00622	0,38108	1,42075	1,36240	1,77074	1,89121	0,65337
0,96356	0,76750	0,89015	0,25337	1,46356	1,42075	1,83603	1,93562	0,46575

Лабораторная работа № 4

ЧИСЛЕННОЕ ИНТЕГРИРОВАНИЕ

Цель работы:

- ознакомиться с численными методами интегрирования функций;
- реализовать алгоритмы численного интегрирования по формуле трапеций и формуле Симпсона в пакете Mathcad;
- для заданной функции $f(x)$ найти приближенные значения определенного интеграла $\int_a^b f(x)dx$ и оценить погрешность вычислений.

Теоретические сведения

В тех случаях, когда не удастся получить интеграл в аналитической форме, либо она очень сложна, применяют численное интегрирование, которое основано на приближенном вычислении площади под кривой, описываемой подынтегральной функцией. В общем виде задача формулируется как нахождение значения

$$I = \int_a^b f(x) dx. \quad (4.1)$$

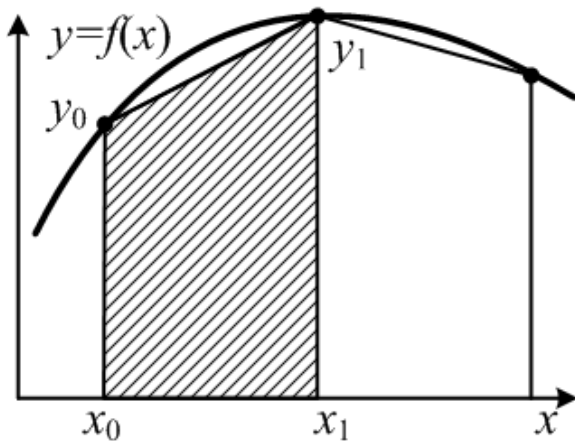


Рис. 4.1. Метод трапеций

При интегрировании **методом трапеций** (рис. 4.1) отрезок $[a, b]$ разбивают на n равных частей:

$$x_0 = a, x_1 = a + h, \dots, x_n = b,$$

$$\text{где } h = \frac{b-a}{n}.$$

Реальную функцию $f(x)$ на каждом интервале $[x_i, x_{i+1}]$ заменяют отрезком прямой, проходящей через точки с координатами (x_i, y_i) и (x_{i+1}, y_{i+1}) (линейная интерполяция).

В этом случае фигура, ограниченная линейным приближением к графику функции и прямыми $x = x_i$, $x = x_{i+1}$, является трапецией. Искомый определенный интеграл определяют как сумму n трапеций высотой h каждая:

$$\int_{x_i}^{x_{i+1}} f(x) dx \approx \frac{b-a}{2n} (y_i + y_{i+1}); \quad (4.2)$$

$$\int_a^b f(x) dx \approx \frac{b-a}{2n} (y_0 + 2y_1 + 2y_2 + \dots + 2y_{n-1} + y_n). \quad (4.3)$$

Функция, реализующая вычисления по формуле трапеций (y – массив значений функции на отрезке $[a, b]$, Δx – шаг):

$$\text{Int}_T(y, \Delta x) := \Delta x \cdot \left(\frac{y_0}{2} + \sum_{i=1}^{\text{last}(y)-1} y_i + \frac{y_{\text{last}(y)}}{2} \right)$$

Очевидно, что чем меньше интервал h , через который задаются значения функции, тем с большей точностью будет вычислен определенный интеграл. Повысить точность результата можно, заменив линейную интерполяцию интерполяцией кривыми более высокого порядка, например параболы (рис. 4.2).

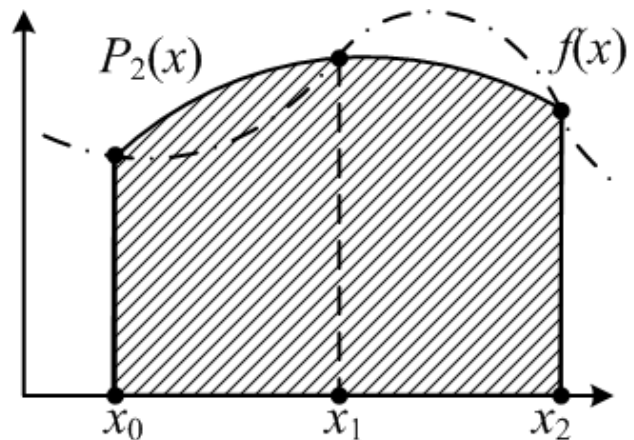


Рис. 4.2. Метод Симпсона

Параболы строят по трем соседним точкам. Для этого отрезок $[a, b]$ разбивают на n равных частей (n – четное). На каждом сдвоенном отрезке $[x_0, x_2], [x_2, x_4], \dots, [x_{n-2}, x_n]$ функцию $f(x)$ заменяют интерполяционным полиномом второй степени $P_2(x)$, построенным по трем узлам интерполяции x_i, x_{i+1}, x_{i+2} :

$$f(x) \approx P_2(x) = y_i + q\Delta y_i + \frac{q(q-1)}{2} \Delta^2 y_i,$$

где $q = \frac{x - x_i}{h}$. Площадь фигуры на сдвоенном интервале $[x_i, x_{i+2}]$:

$$\begin{aligned} S_i &= \int_{x_i}^{x_{i+2}} f(x) dx \approx \int_{x_i}^{x_{i+2}} P_2(x) dx = \left. \begin{array}{l} dx = h \cdot dq; \\ x = x_i \Rightarrow q = 0; \\ x = x_{i+2} \Rightarrow q = 2; \end{array} \right| = \\ &= \int_0^2 \left(y_i + q\Delta y_i + \frac{q(q-1)}{2} \Delta^2 y_i \right) h dq. \end{aligned}$$

С учетом того, что

$$\Delta y_i = y_{i+1} - y_i; \Delta^2 y_i = y_{i+2} - 2y_{i+1} + y_i,$$

получим

$$\int_{x_i}^{x_{i+2}} f(x) dx \approx \frac{b-a}{3n} (y_i + 4y_{i+1} + y_{i+2}). \quad (4.4)$$

Сумма интегралов (4.4) по всем сдвоенным частичным отрезкам дает общую **формулу Симпсона**

$$\int_a^b f(x) dx \approx \frac{b-a}{3n} [y_0 + 4(y_1 + y_3 + \dots + y_{n-1}) + 2(y_2 + y_4 + \dots + y_{n-2}) + y_n]. \quad (4.5)$$

Функция, реализующая вычисления по формуле Симпсона (y – массив значений функции на отрезке $[a, b]$, Δx – шаг):

$$Int_S(y, \Delta x) := \frac{\Delta x}{3} \cdot \left[y_0 + \sum_{i=1}^{N-1} (if(mod(i, 2) = 0, 2, 4) \cdot y_i) + y_N \right]$$

$N := last(y)$

Пример. Найти приближенное значение $\int_0^1 \cos(x) dx$ по формулам трапеций и Симпсона.

Разобьем отрезок интегрирования $[0, 1]$ на $n = 10$ частей.

Зададим исходные данные в пакете Mathcad:

$$\begin{aligned} f(x) &:= \cos(x) & \Delta x &:= 0.1 & n &:= 10 \\ i &:= 0..n & x_i &:= \Delta x \cdot i & y_i &:= f(x_i) \end{aligned}$$

Вычислим заданный определенный интеграл по формулам трапеций и Симпсона:

$$Int_T(y, \Delta x) = 0.84077 \qquad Int_S(y, \Delta x) = 0.841471$$

Результат аналитического решения данной задачи

$$\int_0^1 \cos(x) dx = \sin(1) - \sin(0) = 0,841471.$$

Оценка погрешности численного интегрирования. Обозначим I_h – приближенное значение определенного интеграла, вычисленное с шагом h , I_{2h} – с шагом $2h$.

Погрешность при вычислении интеграла с шагом h удовлетворяет неравенствам:

- для формулы трапеций

$$|R_{mp}| \leq \frac{|I_h - I_{2h}|}{3}; \quad (4.6)$$

- для формулы Симпсона

$$|R_C| \leq \frac{|I_h - I_{2h}|}{15}. \quad (4.7)$$

Постановка задачи

Изучить численные методы вычисления значения определенного интеграла.

Вычислить определенный интеграл (согласно заданному варианту) по формулам трапеций и Симпсона при $n = 12$.

Оценить погрешность полученных результатов.

Содержание отчета

Отчет о лабораторной работе должен содержать:

- постановку задачи;
- пользовательские функции Mathcad, определяющие значение интеграла по формулам трапеций и Симпсона;
- приближенные значения заданного определенного интеграла;
- оценку погрешности полученных результатов.

Контрольные вопросы

1. В каких случаях не обойтись без численного интегрирования?
2. Сравните методы интегрирования по формулам трапеций и Симпсона: суть методов, особенности, погрешности.
3. От чего зависит погрешность численного интегрирования?
4. Почему в формуле Симпсона число отрезков разбиения интервала интегрирования должно быть четным?
5. Приведите примеры функций, для которых погрешность формулы трапеций меньше, чем погрешность формулы Симпсона.
6. Приведите примеры функций, для которых интегрирование по формуле Симпсона даст точный результат.
7. Можно ли применять для численного интегрирования интерполяционные полиномы третьего и более высоких порядков? Ответ обоснуйте.

Варианты заданий

Номер варианта	Интеграл
1	$\int_0^{1,2} \frac{1,5x^2 + x}{x^5 + 1} dx$
2	$\int_2^{4,4} \frac{x - x^3}{x^4 + 2} dx$
3	$\int_1^{2,2} \frac{2,2x + 1,7}{3,1x^3 + 9,3x} dx$
4	$\int_0^{1,8} \frac{x^2 - 4,1}{x^4 + 1} dx$
5	$\int_0^{1,2} \frac{3,5x^2 + x}{x^4 + 2} dx$
6	$\int_2^{4,4} \frac{1 - 3,7x}{x^3 + x} dx$
7	$\int_1^{2,8} \frac{x^3 + 4,5}{x^4 + 5,4} dx$
8	$\int_1^{3,4} \frac{2,4 + x^2}{x^3 + 8,1} dx$
9	$\int_1^{2,2} \frac{5,2x + 4,8}{4,5x^2 + 4,1} dx$
10	$\int_0^{1,8} \frac{x^2 - 3,7}{x^5 + 1} dx$
11	$\int_2^{3,2} \frac{3,3x - 2,8}{4,5x^3 + 7,4} dx$
12	$\int_0^{1,2} \frac{x - 1,8}{x^4 + 5,1} dx$

Номер варианта	Интеграл
13	$\int_2^{3,2} \frac{3,3x - 2,8}{4,5x^3 + 7,4} dx$
14	$\int_2^{3,2} \frac{1,6x - 3,8}{1,2x^3 + 1,7} dx$
15	$\int_1^{2,2} \frac{x^2 - 4,9}{x^3 + 1,8} dx$
16	$\int_1^{3,4} \frac{x^2 + 1,5x}{x^3 + 7,1} dx$
17	$\int_1^{2,2} \frac{x^2 + 3,8}{x^3 + 5,1} dx$
18	$\int_2^{4,4} \frac{x^2 - 1,3}{1,2x^3 + 1} dx$
19	$\int_0^{1,2} \frac{3,8x - 7,5}{2,5x^2 + 2} dx$
20	$\int_1^{2,2} \frac{1,8x^2 - 1}{x^3 + 1,2} dx$
21	$\int_2^{3,2} \frac{3,1x + 2,1}{x^3 + 2} dx$
22	$\int_1^{3,4} \frac{x^2 - 6,1}{x^4 - 2,4} dx$
23	$\int_2^{4,4} \frac{1,8 + x}{x^3 + 4,8} dx$
24	$\int_0^{2,2} \frac{x^2 + 1,9}{x^4 + 3,1} dx$

Лабораторная работа № 5

АППРОКСИМАЦИЯ ТАБЛИЧНЫХ ДАННЫХ КРИВЫМИ

Цель работы:

- ознакомиться с методами теории приближений таблично заданных функций;
- реализовать алгоритм полиномиальной аппроксимации по методу наименьших квадратов в пакете Mathcad;
- найти по табличным данным аппроксимирующие кривые в виде линейного и квадратичного полиномов;
- оценить среднеквадратичную погрешность аппроксимации.

Теоретические сведения

Существует два основных подхода к описанию (приближению) табличных данных с помощью элементарных функций. В первом – требуют, чтобы аппроксимирующая кривая (возможно, кусочно-гладкая) проходила через все точки, заданные таблицей (рис. 5.1, а). Эту задачу решают методами интерполяции. Во втором – данные аппроксимируют простой функцией, применимой во всем диапазоне табличных данных, но не обязательно проходящей через все точки (рис. 5.1, б). Выбор того или иного подхода зависит от объема данных и от информации об их точности. Например, если экспериментальные данные измерены с заметной погрешностью, то интерполяция нецелесообразна; аппроксимация позволит сгладить ошибки эксперимента.

Аппроксимацию данных называют подгонкой кривой, которую требуется провести так, чтобы ее отклонения от табличных данных были минимальными, т.е. функцию $f(x)$, заданную своими значениями в дискретных узлах $y_i = f(x_i)$, $i = 1, \dots, n$, заменяют в некотором смысле близкой функцией $g(x)$:

$$\|f(x_i) - g(x_i)\| < \varepsilon,$$

где ε – точность аппроксимации.

С учетом этого задача приближения функции может быть сформулирована следующим образом: найти такую функцию

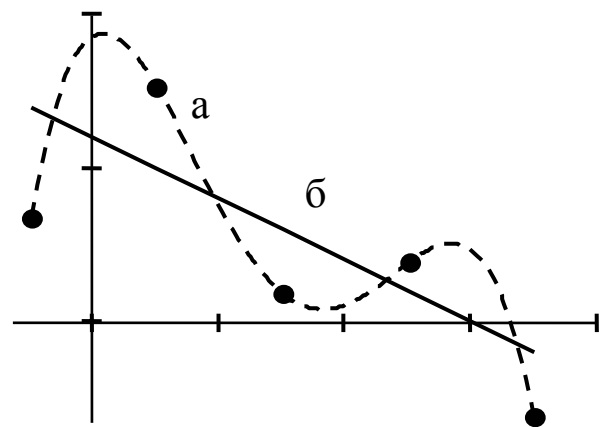


Рис. 5.1. Интерполяция (а) и аппроксимация (б) данных

$g(x)$ заданного вида, чтобы расстояние в n -мерном пространстве между точками $M(y_1, y_2, \dots, y_n)$ и $\bar{M}(g(x_1), g(x_2), \dots, g(x_n))$ было наименьшим.

Отметим, что в качестве меры близости (или расстояния) нельзя брать сумму отклонений $\delta_i = f(x_i) - g(x_i)$, так как отклонения с противоположными знаками компенсируют друг друга (рис. 5.2, а). Если учитывать абсолютные значения отклонений $|\delta_i|$, то возникают проблемы при нахождении минимума с использованием необходимого условия существования экстремума, так как для функции «модуль» в точке минимума производная не существует (рис. 5.2, б).

Поэтому требуется свести к минимуму сумму квадратов разностей между значениями аппроксимирующей функции $g(x_i)$ и табличными значениями y_i . Такой метод подгонки называется **методом наименьших квадратов**.

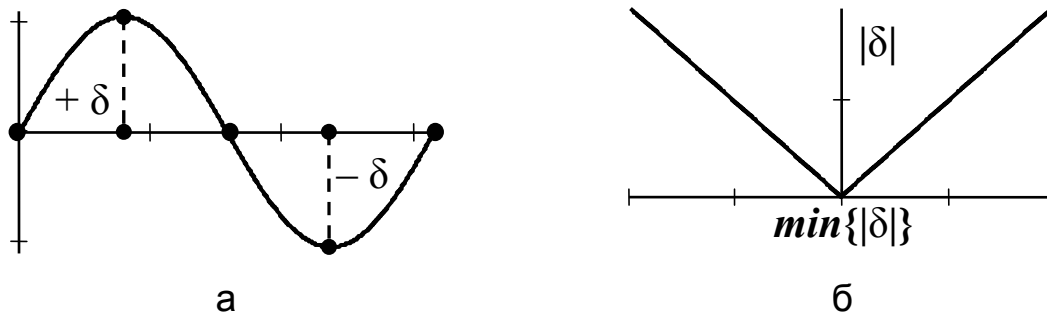


Рис. 5.2. Пояснение выбора меры расстояния между функцией $f(x)$ и аппроксимирующей кривой $g(x)$

Пусть в таблице заданы $n + 1$ точки $(x_0, y_0), (x_1, y_1), \dots, (x_n, y_n)$ и требуется найти аппроксимирующую кривую $g(x)$ в диапазоне $[x_0, x_n]$.

Функцию $g(x)$ нужно выбирать с учетом специфики табличных данных, под которой понимается их периодичность, экспоненциальный или логарифмический характер, свойство симметрии и наличие асимптотики. Иногда таблицу разбивают на несколько частей и подбирают отдельную аппроксимирующую кривую для каждой части. Это оправдано в тех случаях, когда есть основание полагать, что табличные данные соответствуют разным физическим состояниям исследуемого объекта (например, переход от устойчивого состояния системы к неустойчивому).

Для аппроксимации данных можно использовать следующие элементарные функции:

$$y = \sum_{i=0}^m a_i x^i \quad (m \ll n); \quad y = ax^m; \quad y = ae^{mx};$$

$y = \frac{cx}{ax+b}$; $y = a\frac{1}{x} + b$; $y = a \ln(x) + b$. Обычно $g(x)$ выбирают в виде линейной комбинации подходящих функций:

$$g(x) = c_1 g_1(x) + c_2 g_2(x) + \dots + c_k g_k(x). \quad (5.1)$$

Погрешность аппроксимации в каждой табличной точке

$$\varepsilon_i = y_i - g(x_i).$$

Тогда сумма квадратов погрешностей

$$E = \sum_{i=0}^n [y_i - g(x_i)]^2. \quad (5.2)$$

С учетом (5.1) условие $\min\{E\}$ определяется уравнениями

$$\frac{\partial E}{\partial c_1} = \frac{\partial E}{\partial c_2} = \dots = \frac{\partial E}{\partial c_k} = 0.$$

Это условие эквивалентно системе уравнений

$$\begin{cases} \frac{\partial E}{\partial c_1} = 2 \sum_{i=0}^n [c_1 g_1(x_i) + \dots + c_k g_k(x_i) - y_i] g_1(x_i) = 0; \\ \frac{\partial E}{\partial c_2} = 2 \sum_{i=0}^n [c_1 g_1(x_i) + \dots + c_k g_k(x_i) - y_i] g_2(x_i) = 0; \\ \vdots \\ \frac{\partial E}{\partial c_k} = 2 \sum_{i=0}^n [c_1 g_1(x_i) + \dots + c_k g_k(x_i) - y_i] g_k(x_i) = 0 \end{cases} \quad (5.3)$$

или в матричной форме

$$\begin{bmatrix} \sum_{i=0}^n g_1^2(x_i) & \dots & \sum_{i=0}^n g_1(x_i)g_k(x_i) \\ \sum_{i=0}^n g_2(x_i)g_1(x_i) & \dots & \sum_{i=0}^n g_2(x_i)g_k(x_i) \\ \vdots & \ddots & \vdots \\ \sum_{i=0}^n g_k(x_i)g_1(x_i) & \dots & \sum_{i=0}^n g_k^2(x_i) \end{bmatrix} \begin{bmatrix} c_1 \\ c_2 \\ \vdots \\ c_k \end{bmatrix} = \begin{bmatrix} \sum_{i=0}^n g_1(x_i)y_i \\ \sum_{i=0}^n g_2(x_i)y_i \\ \vdots \\ \sum_{i=0}^n g_k(x_i)y_i \end{bmatrix}. \quad (5.4)$$

Так как элементы матрицы в левой части (5.4) и вектора столбца в правой части определяются табличными данными, то система (5.3) k линейных уравнений с k неизвестными может быть решена.

Полиномиальная аппроксимация. Пусть в результате эксперимента получено $n + 1$ значение функции $y_i = f(x_i)$. Требуется аппрок-

симировать эту функцию полиномом степени m ($m < n$):

$$g(x) = P_m(x) = a_0 + a_1x + a_2x^2 + \dots + a_mx^m.$$

Согласно методу наименьших квадратов необходимо найти те значения параметров a_0, a_1, \dots, a_m , при которых сумма квадратов отклонений минимальна:

$$E = \sum_{i=0}^n [P_m(x_i) - y_i]^2 = \sum_{i=0}^n [a_0 + a_1x_i + \dots + a_mx_i^m - y_i]^2.$$

Используя необходимые условия существования экстремума функции нескольких переменных $\frac{\partial E}{\partial a_k} = 0, k = 0, \dots, m$, получаем систему уравнений для определения неизвестных a_0, a_1, \dots, a_m :

$$\sum_{i=0}^n [a_0 + a_1x_i + \dots + a_mx_i^m - y_i]^2 x_i^k = 0, k = 0, \dots, m.$$

Рассмотрим частные случаи нахождения приближающей функции $g(x)$ в виде линейного $P_1(x)$ и квадратичного $P_2(x)$ полиномов.

Пусть $m = 1$, т.е. функцию аппроксимируем полиномом первой степени $P_1(x) = a_0 + a_1x$. Система уравнений для вычисления параметров a_0, a_1 имеет следующий вид:

$$\begin{cases} a_0(n+1) + a_1 \sum_{i=0}^n x_i = \sum_{i=0}^n y_i; \\ a_0 \sum_{i=0}^n x_i + a_1 \sum_{i=0}^n x_i^2 = \sum_{i=0}^n x_i y_i. \end{cases} \quad (5.5)$$

Обозначим \mathbf{A} – матрица коэффициентов системы (5.5), \mathbf{B} – столбец свободных членов:

$$\mathbf{A} = \begin{bmatrix} n+1 & \sum_{i=0}^n x_i \\ \sum_{i=0}^n x_i & \sum_{i=0}^n x_i^2 \end{bmatrix}; \quad \mathbf{B} = \begin{bmatrix} \sum_{i=0}^n y_i \\ \sum_{i=0}^n x_i y_i \end{bmatrix}.$$

Тогда решение системы линейных уравнений (5.5) находим как

$$\begin{bmatrix} a_0 \\ a_1 \end{bmatrix} = \mathbf{A}^{-1} \mathbf{B}.$$

Если функцию аппроксимируем полиномом второй степени $P_2(x) = a_0 + a_1x + a_2x^2$, то система уравнений для вычисления па-

параметров a_0, a_1, a_2 имеет следующий вид:

$$\begin{cases} a_0(n+1) + a_1 \sum_{i=0}^n x_i + a_2 \sum_{i=0}^n x_i^2 = \sum_{i=0}^n y_i; \\ a_0 \sum_{i=0}^n x_i + a_1 \sum_{i=0}^n x_i^2 + a_2 \sum_{i=0}^n x_i^3 = \sum_{i=0}^n x_i y_i; \\ a_0 \sum_{i=0}^n x_i^2 + a_1 \sum_{i=0}^n x_i^3 + a_2 \sum_{i=0}^n x_i^4 = \sum_{i=0}^n x_i^2 y_i. \end{cases} \quad (5.6)$$

В этом случае

$$\mathbf{A} = \begin{bmatrix} n+1 & \sum_{i=0}^n x_i & \sum_{i=0}^n x_i^2 \\ \sum_{i=0}^n x_i & \sum_{i=0}^n x_i^2 & \sum_{i=0}^n x_i^3 \\ \sum_{i=0}^n x_i^2 & \sum_{i=0}^n x_i^3 & \sum_{i=0}^n x_i^4 \end{bmatrix}; \quad \mathbf{B} = \begin{bmatrix} \sum_{i=0}^n y_i \\ \sum_{i=0}^n x_i y_i \\ \sum_{i=0}^n x_i^2 y_i \end{bmatrix}.$$

Решение системы линейных уравнений (5.6):

$$\begin{bmatrix} a_0 \\ a_1 \\ a_2 \end{bmatrix} = \mathbf{A}^{-1} \mathbf{B}.$$

Вычисления коэффициентов систем (5.5) и (5.6) удобно проводить в пакете Mathcad.

Функция Mathcad, определяющая матрицу коэффициентов \mathbf{A} :

$$A(x, m) := \left| \begin{array}{l} n \leftarrow \text{last}(x) \\ \text{for } j \in 0..m \\ \quad \text{for } i \in 0..m \\ \qquad a_{i,j} \leftarrow \sum_{k=0}^{n-1} (x_k)^{i+j} \end{array} \right| a$$

Аргументы функции: x – массив координат узловых точек; m – степень аппроксимирующего полинома.

Функция Mathcad, определяющая матрицу коэффициентов **B**:

$$B(x, y, m) := \left| \begin{array}{l} n \leftarrow \text{last}(x) \\ \text{for } i \in 0..m \\ b_i \leftarrow \sum_{k=0}^{n-1} [y_k \cdot (x_k)^i] \\ b \end{array} \right.$$

Аргументы функции: y – массив значений функции в узловых точках, остальные – те же, что и для функции $A(x, m)$.

Пример. Функция задана таблично.

x_i	-3	-1	0	1	2	3	4
y_i	2,9	1,0	-0,2	-1,5	-0,4	0,5	2,0

Требуется найти полиномы первой и второй степеней, аппроксимирующие заданную функцию.

Зададим исходные данные в пакете Mathcad:

$$i := 0..6$$

Найдем матрицу коэффициентов и столбец свободных членов системы (5.5):

$$A1 := A(x, 1)$$

$$A1 = \begin{pmatrix} 10 & 8.4 \\ 8.4 & 41.94 \end{pmatrix}$$

$$B1 := B(x, y, 1)$$

$$B1 = \begin{pmatrix} 6.383 \\ -0.848 \end{pmatrix}$$

$$x_i :=$$

-3
-1
0
1
2
3
4

$$y_i :=$$

2.9
1.0
-0.2
-1.5
-0.4
0.5
2.0

Вычислим параметры полинома первой степени $P_1(x)$:

$$\begin{pmatrix} a_0 \\ a_1 \end{pmatrix} := A1^{-1} \cdot B1$$

$$\begin{pmatrix} a_0 \\ a_1 \end{pmatrix} = \begin{pmatrix} 0.788 \\ -0.178 \end{pmatrix}$$

Определим функцию для расчета значений полинома $P_1(x)$:

$$p1(x) := a_0 + a_1 \cdot x$$

Найдем матрицу коэффициентов и столбец свободных членов системы (5.6):

$$A2 := A(x, 2) \qquad B2 := B(x, y, 2)$$

$$A2 = \begin{pmatrix} 10 & 8.4 & 41.94 \\ 8.4 & 41.94 & 73.584 \\ 41.94 & 73.584 & 437.306 \end{pmatrix} \qquad B2 = \begin{pmatrix} 6.383 \\ -0.848 \\ 61.824 \end{pmatrix}$$

Вычислим параметры полинома второй степени $P_2(x)$:

$$\begin{pmatrix} b_0 \\ b_1 \\ b_2 \end{pmatrix} := A2^{-1} \cdot B2 \qquad \begin{pmatrix} b_0 \\ b_1 \\ b_2 \end{pmatrix} = \begin{pmatrix} 0.163 \\ -0.388 \\ 0.191 \end{pmatrix}$$

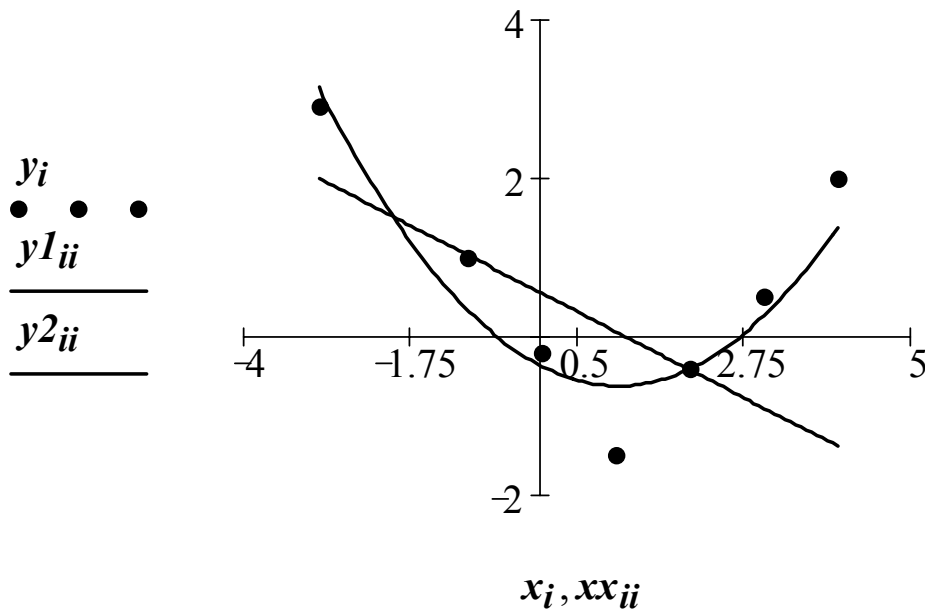
Определим функцию для расчета значений полинома $P_1(x)$:

$$p2(x) := b_0 + b_1 \cdot x + b_2 \cdot x^2$$

Построим графики линейного $P_1(x)$ и квадратичного $P_2(x)$ полиномов, совместив их с табличными данными:

$$ii := 0 .. 100 \qquad xx_{ii} := \min(x) + \frac{\max(x) - \min(x)}{100} \cdot ii$$

$$y1_{ii} := p1(xx_{ii}) \qquad y2_{ii} := p2(xx_{ii})$$



Найдем отклонения исходных данных от вычисленных значений:

$$\sum_i (y_i - p1(x_i))^2 = 12.374 \qquad \sum_i (y_i - p2(x_i))^2 = 2.831$$

Постановка задачи

Изучить метод наименьших квадратов для полиномиальной аппроксимации результатов наблюдений.

Найти многочлены первой и второй степени, аппроксимирующие функцию, заданную таблично.

Оценить суммы квадратов отклонений табличных данных от вычисленных значений с помощью аппроксимирующих кривых.

Содержание отчета

Отчет о лабораторной работе должен содержать:

- постановку задачи;
- системы линейных уравнений (5.5), (5.6) для определения параметров аппроксимирующих полиномов первой и второй степеней;
- пользовательские функции Mathcad, определяющие матрицы коэффициентов \mathbf{A} и столбцы свободных членов \mathbf{B} систем (5.5), (5.6);
- оценки коэффициентов полиномов $P_1(x)$ и $P_2(x)$;
- графическое представление полученных результатов (графики аппроксимирующих полиномов, совмещенные с табличными данными в одних координатных осях);
- оценку квадратичной погрешности полученных результатов.

Контрольные вопросы

1. Что является исходными данными и что требуется найти в задаче аппроксимации?
2. В чем сходство и отличие задач аппроксимации и интерполяции данных?
3. Приведите примеры практических задач, приводящих к аппроксимации функций.
4. Как выбрать вид аппроксимирующей функции?
5. Как выбрать степень аппроксимирующего полинома?
6. В чем заключается идея метода наименьших квадратов? О каких квадратах идет речь?
7. Можно ли минимизировать не сумму квадратов, а сумму модулей отклонений аппроксимирующей кривой от табличных данных? Ответ обоснуйте.
8. Можно ли по семи заданным точкам построить полиномы третьей, четвертой, пятой и т.д. степеней? Если – да, то как? Если – нет, то почему?

Варианты заданий

1	x_i	0	1	2	3	4	5	6
	y_i	0,2	0,6	1,0	1,2	1,4	1,6	1,7
2	x_i	-2	-1	0	1	2	3	4
	y_i	3,1	2,8	2,5	2,0	1,7	2,2	2,9
3	x_i	-6	-4	-3	-1	0	1	3
	y_i	2,5	1,2	0,4	-0,5	-1,3	-0,2	1,1
4	x_i	0	1	2	3	4	5	6
	y_i	0,5	0,8	1,3	1,7	1,9	2,5	2,2
5	x_i	-3	-2	-1	0	1	2	4
	y_i	1,7	1,2	1,0	0,5	-0,2	0,5	0,8
6	x_i	-1	0	1	2	3	4	5
	y_i	3,1	2,8	2,4	2,1	1,9	2,2	2,6
7	x_i	1	2	3	4	5	6	7
	y_i	1,8	1,2	0,2	-0,9	-1,9	0,4	2,4
8	x_i	-2	-1	0	1	2	3	4
	y_i	1,7	1,9	2,4	2,7	3,1	3,1	2,5
9	x_i	0	1	2	3	4	5	6
	y_i	1,4	2,0	2,3	2,9	2,5	2,3	2,0
10	x_i	-1	0	1	2	3	4	5
	y_i	-1,8	-1,5	-1,1	-1,3	-1,4	-1,6	-1,9
11	x_i	-3	-2	-1	0	1	3	4
	y_i	1,0	1,7	3,3	5,1	4,6	3,0	1,9
12	x_i	0	1	2	3	4	5	6
	y_i	2,1	3,0	3,4	3,7	3,2	2,9	1,1
13	x_i	-2	-1	0	1	2	3	4
	y_i	-0,3	0,5	0,8	1,8	0,8	0,4	0,0
14	x_i	0	1	2	3	4	5	6
	y_i	1,6	1,9	2,3	2,5	2,8	3,4	2,5
15	x_i	-2	-1	0	1	2	3	4
	y_i	0,3	-0,5	-1,5	-0,5	-0,1	0,2	1,2

Окончание вариантов заданий

16	x_i	-3	-2	-1	0	1	2	4
	y_i	4,8	4,2	3,7	3,4	3,0	3,6	5,0
17	x_i	0	1	2	3	4	5	6
	y_i	3,5	3,2	2,9	2,1	3,0	3,4	3,9
18	x_i	-1	0	1	2	3	4	5
	y_i	-6,1	-5,8	-5,2	-4,8	-4,5	-5,0	-5,6
19	x_i	-2	-1	0	1	2	3	4
	y_i	1,1	0,2	-0,4	-1,0	-1,4	-1,0	-0,2
20	x_i	0	1	2	3	4	5	6
	y_i	-1,2	-0,5	-0,2	0,3	-0,7	-1,1	-1,4
21	x_i	-3	-1	0	1	3	4	6
	y_i	1,7	3,3	5,1	6,6	5,6	4,0	3,5
22	x_i	0	1	2	3	4	5	6
	y_i	1,7	1,9	2,5	2,9	3,4	2,8	2,2
23	x_i	-4	-3	-2	-1	0	1	3
	y_i	-1,8	-0,5	-0,2	0,5	1,0	1,2	0,3
24	x_i	-1	0	1	2	3	4	5
	y_i	3,1	4,5	4,9	5,7	5,2	4,2	3,0
25	x_i	-2	-1	0	1	2	3	4
	y_i	-0,3	0,5	1,5	0,5	0,3	-0,2	-1,2
26	x_i	-6	-5	-4	-3	-2	-1	0
	y_i	-2,4	-3,5	-4,1	-3,4	-2,3	-1,5	-0,7
27	x_i	-3	-2	-1	0	1	2	3
	y_i	6,4	4,1	3,5	2,6	1,3	-0,2	1,7
28	x_i	-1	0	1	2	3	4	5
	y_i	1,6	3,2	4,8	5,5	4,1	3,7	2,4
29	x_i	0	1	2	3	4	5	6
	y_i	-1,7	-0,5	-0,1	0,9	1,8	0,8	0,3
30	x_i	-2	-1	0	1	2	3	4
	y_i	1,3	1,8	2,5	3,1	3,7	2,2	1,9

Лабораторная работа № 6

МЕТОДЫ РЕШЕНИЯ ОБЫКНОВЕННЫХ ДИФФЕРЕНЦИАЛЬНЫХ УРАВНЕНИЙ. ЗАДАЧА КОШИ

Цель работы:

- ознакомиться с одношаговыми и многошаговыми методами решения дифференциальных уравнений с начальными условиями;
- реализовать алгоритмы одношаговых и многошаговых методов решения задачи Коши в пакете Mathcad;
- вычислить значения приближенного решения заданного дифференциального уравнения первого порядка.

Теоретические сведения

Любая задача проектирования, связанная с расчетом потоков энергии или движения тел, сводится к решению дифференциальных уравнений (ДУ). В зависимости от числа независимых переменных ДУ разделяют на две категории:

- *обыкновенные*, содержащие одну независимую переменную и производные по ней;
- *в частных производных*, содержащие несколько независимых переменных и частные производные по ним.

Порядок ДУ определяется максимальным порядком производной. ДУ первого порядка – это соотношение вида

$$F\left(x, y, \frac{dy}{dx}\right) = 0. \quad (6.1)$$

Решением ДУ на интервале I называется непрерывно дифференцируемая функция $y = \varphi(x)$, превращающая уравнение в тождество на интервале I . Решения ДУ: *общее* (семейство подобных кривых, отличающихся на константу) и *частное* (одно из множества общих решений, удовлетворяющее дополнительным условиям, определяющим значение функции и/или ее производных при некоторых значениях независимой переменной). Если эти условия задают при одном значении переменной, то получают **задачу с начальными условиями** или **задачу Коши**. Если же условия задают при двух или более значениях независимой переменной, то задачу называют **краевой**, а условия – граничными.

Задачу Коши формулируют следующим образом. Пусть дано ДУ $y'(x) = f(x, y)$ и начальное условие $y(x_0) = y_0$. Требуется найти

функцию $y(x)$, удовлетворяющую как данному уравнению, так и начальному условию. С помощью численных методов можно получить приближенные значения функции-решения $y(x_i)$ в заданных точках x_i .

Существуют две группы методов решения задачи Коши:

– **одношаговые**, в которых для нахождения следующей точки на кривой $y = \varphi(x)$ требуется информация только об одном предыдущем шаге (методы Эйлера и Рунге - Кутты);

– **многошаговые (прогноза и коррекции)**, в которых для отыскания следующей точки решения требуется информация более чем об одной из предыдущих точек (метод Адамса и др.).

Метод Эйлера – простейший метод решения задачи Коши. Его точность невелика, поэтому на практике им пользуются редко. Однако на основе этого метода легче понять алгоритм других, более эффективных методов.

Метод Эйлера основан на разложении функции $y(x)$ в ряд Тейлора в окрестности точки x_0 :

$$y(x_0 + h) = y(x_0) + hy'(x_0) + \frac{1}{2!}h^2 y''(x_0) + \dots \quad (6.2)$$

Если шаг h мал, то члены разложения (6.2), содержащие h во второй и более высоких степенях, являются малыми более высоких порядков и ними можно пренебречь. Тогда

$$y(x_0 + h) = y(x_0) + hy'(x_0),$$

где $y'(x_0)$ находим из ДУ, подставив в него начальное условие

$$y'(x_0) = f(x_0, y_0).$$

Этот процесс можно продолжить, используя соотношение

$$y_{n+1} = y_n + hf(x_n, y_n), \quad n = 1, 2, \dots \quad (6.3)$$

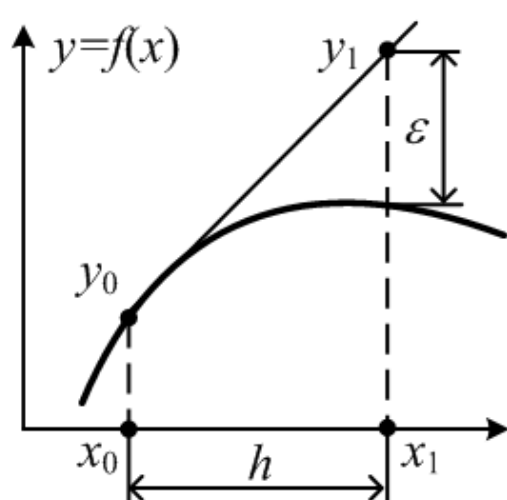


Рис. 6.1. Метод Эйлера

Ошибка метода имеет порядок h^2 .

Графически метод Эйлера показан на рис. 6.1. Угол наклона касательной к истинной кривой изменяется в соответствии с изменением независимой переменной x . В точке $(x_0 + h)$ наклон касательной не такой, каким он был в x_0 .

Следовательно, при сохранении на всем интервале h начального угла наклона касательной к функции, равно $\arctg[y'(x_0)]$, в результате вычислений вносится погрешность ε .

Точность метода Эйлера можно повысить, улучшив аппроксимацию производной $y'(x)$. В **первом модифицированном методе Эйлера** для вычисления следующей точки используют производную, вычисленную в точке псевдосередины интервала:

$$y_{n+1} = y_n + hf \left(x_{n+\frac{1}{2}}, y_{n+\frac{1}{2}} \right), \quad (6.4)$$

где $x_{n+\frac{1}{2}} = x_n + \frac{h}{2}$, $y_{n+\frac{1}{2}} = y_n + \frac{h}{2} y'_n$.

Во **втором модифицированном методе Эйлера** применяют среднее значение производных в начале и в конце шага:

$$y_{n+1} = y_n + h \frac{y'_n + y'_{n+1}}{2}, \quad (6.5)$$

где $y'_{n+1} = f(x_n + h, y_n + hy'_n)$.

Ошибка на каждом шаге при использовании модифицированных методов Эйлера имеет порядок h^3 .

Более высокая точность может быть достигнута вследствие лучшей аппроксимации производной путем сохранения большего числа членов ряда Тейлора (6.2). Эта идея лежит в основе **методов Рунге - Кутта**. Наиболее распространенный – метод четвертого порядка (он содержит члены ряда Тейлора, включая h^4); в этом случае ошибка на каждом шаге имеет порядок h^5 .

Расчеты при использовании этого метода выполняют по формуле

$$y_{n+1} = y_n + \Delta y_n, \quad \Delta y_n = \frac{1}{6}(K_0 + 2K_1 + 2K_2 + K_3), \quad (6.6)$$

где $K_0 = hf(x_n, y_n)$, $K_1 = hf \left(x_n + \frac{h}{2}, y_n + \frac{K_0}{2} \right)$,

$$K_2 = hf \left(x_n + \frac{h}{2}, y_n + \frac{K_1}{2} \right), \quad K_3 = hf(x_n + h, y_n + K_2).$$

Метод Эйлера и его модификации, по сути, являются методами Рунге - Кутта первого и второго порядков.

Таким образом, всем **одношаговым методам** присущи определенные общие черты:

- чтобы получить решение в новой точке, надо иметь данные лишь в одной предыдущей точке;
- в основе методов лежит разложение функции в ряд Тейлора, в

котором сохраняются члены, содержащие h в степени до k включительно. Число k называется порядком метода. Погрешность на шаге имеет порядок h^{k+1} ;

– не требуется определение производных: вычисляют только функцию $f(x, y)$, однако могут потребоваться ее значения в нескольких промежуточных точках;

– можно изменять величину шага h .

Функция, реализующая метод Рунге - Кутты четвертого порядка, для ДУ первого порядка:

```

RungeKutta (x0, y0, f, h, N) :=
  z0,0 ← x0
  z0,1 ← y0
  for i ∈ 1..N
    zi,0 ← x0 + h · i
    x ← zi-1,0
    y ← zi-1,1
    k1 ← h · f(x, y)
    k2 ← h · f(x + h/2, y + k1/2)
    k3 ← h · f(x + h/2, y + k2/2)
    k4 ← h · f(x + h, y + k3)
    Δy ← (k1 + 2 · k2 + 2 · k3 + k4) / 6
    zi,1 ← y + Δy
  z
  
```

Аргументы функции: x_0 – начальное значение независимой переменной; y_0 – значение решения в точке x_0 ; f – имя функции, стоящей в правой части ДУ; h – шаг приращения переменной x ; N – число сетки, на которой ищется решение ДУ. Функция возвращает матрицу, состоящую из двух столбцов: первый столбец – значения аргумента x_i , второй – значения решения ДУ $y(x_i)$.

Пример. Используя метод Рунге - Кутта, вычислить значения y_1 , y_2 , y_3 приближенного решения ДУ

$$y' = 1,2(x + y)^2 + 0,5(x - y)$$

с начальным условием $y(0) = 0$ и шагом сетки $h = 0,1$.

Зададим исходные данные в пакете Mathcad:

$$f(x, y) := 1.2 \cdot (x + y)^2 + 0.5 \cdot (x - y)$$

$$x_0 := 0 \quad y_0 := 0 \quad h := 0.1$$

Найдем значения y_1 , y_2 , y_3 приближенного решения ДУ с помощью функции, реализующей вычислительную схему метода Рунге - Кутта четвертого порядка:

$$R := \text{RungeKutta}(x_0, y_0, f, h, 3)$$

$$R = \begin{pmatrix} 0 & 0 \\ 0.1 & 0.0029 \\ 0.2 & 0.0131 \\ 0.3 & 0.0335 \end{pmatrix} \quad \begin{array}{l} x_1 := R_{1,0} \\ x_2 := R_{2,0} \\ x_3 := R_{3,0} \end{array} \quad \begin{array}{l} y_1 := R_{1,1} \\ y_2 := R_{2,1} \\ y_3 := R_{3,1} \end{array}$$

Результаты расчета:

$$y_1 = 0.0029 \quad y_2 = 0.0131 \quad y_3 = 0.0335$$

В **многошаговых методах** используют две формулы – прогноза и коррекции. До начала вычислений необходимо получить посредством какого-либо одношагового метода (чаще всего метода Рунге - Кутта) исходные данные – предыдущие значения решения. Схемы алгоритмов для всех многошаговых методов примерно одинаковы, а сами методы отличаются только формулами.

На первом шаге алгоритма по формуле прогноза и исходным значениям определяют $y_{n+1}^{(0)}$.

Подставив прогнозируемое значение $y_{n+1}^{(0)}$ в заданное ДУ $y'(x) = f(x, y)$, находят производную $y'_{n+1}^{(0)} = f(x_{n+1}, y_{n+1}^{(0)})$.

Величину $y'_{n+1}^{(0)}$ подставляют в формулу коррекции для вычисления уточненного значения $y_{n+1}^{(j+1)}$, которое используют для получения более точного значения производной $y'_{n+1}^{(j+1)} = f(x_{n+1}, y_{n+1}^{(j+1)})$.

Если это значение производной недостаточно близко к предыдущему, то его вводят в формулу коррекции и итерационный процесс продолжается ($j = 0, 1, 2, \dots$).

Если оценка производной изменяется в допустимых пределах, то $y'_{n+1}^{(j+1)}$ используют для вычисления окончательного значения y_{n+1} .

Обычно при выводе формул прогноза и коррекции решение ДУ рассматривают как процесс приближенного интегрирования, а сами формулы получают с помощью конечно-разностных методов.

Если ДУ $y'(x) = f(x, y)$ проинтегрировать в интервале значений от x_n до x_{n+1} , то результат имеет вид

$$y(x_{n+1}) - y(x_n) = \int_{x_n}^{x_{n+1}} f(x, y) dx.$$

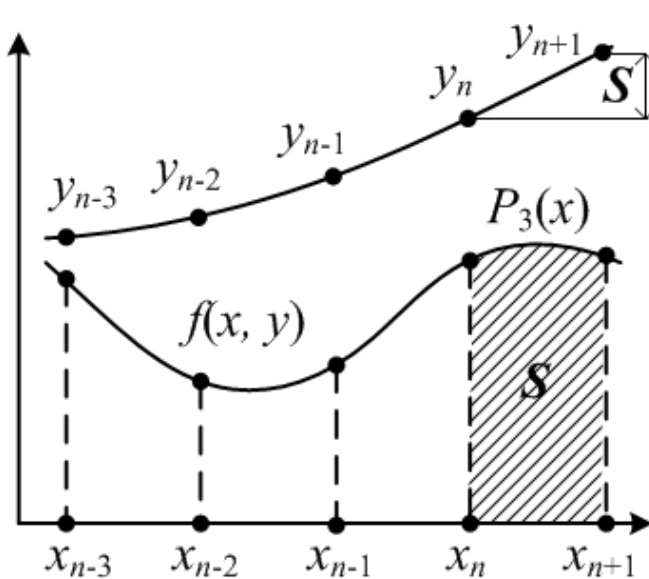


Рис. 6.2. Метод Адамса

Этот интеграл нельзя вычислить непосредственно, так как зависимость $y(x)$ неизвестна. Можно экстраполировать производную искомой функции $f(x, y)$ по четырем известным точкам полиномом третьей степени $P_3(x)$ (рис. 6.2). Тогда площадь S под полиномом на интервале $[x_n, x_{n+1}]$ будет равна приращению Δy искомой функции на следующем шаге.

В методе Адамса формула прогноза получена путем интегрирования второй интерполяционной формулы Ньютона (так как требуется найти значение интерполяционного полинома в точке x_{n+1} , близкой к концу таблицы):

$$y_{n+1} = y_n + \frac{1}{24} h (55 y'_n - 59 y'_{n-1} + 37 y'_{n-2} - 9 y'_{n-3}). \quad (6.7)$$

Формула коррекции

$$y_{n+1} = y_n + \frac{1}{24} h (9 y'_{n+1} - 19 y'_n - 5 y'_{n-1} + y'_{n-2}). \quad (6.8)$$

Экстраполяционный метод Адамса имеет четвертый порядок точности, т.е. погрешность на каждом шаге – величина порядка h^5 .

Для начала вычислений по (6.7) нужны четыре значения – y_0, y_1, y_2, y_3 . Значение y_0 задается начальным условием, а значения y_1, y_2, y_3 вычисляют по методу Рунге - Кутты.

Пример. Используя результаты, полученные в предыдущем примере, вычислить значения y_4, y_5 приближенного решения ДУ по методу Адамса.

Найдем координаты узлов сетки x_4 и x_5 :

$$i := 4..5 \qquad x_i := x_0 + h \cdot i$$

С учетом того, что значения y_0, \dots, y_3 известны, вычислим значения y_4, y_5 по формуле прогноза (6.7):

$$y_i := y_{i-1} + \frac{h}{24} \cdot (55 f(x_{i-1}, y_{i-1}) - 59 f(x_{i-2}, y_{i-2}) + 37 f(x_{i-3}, y_{i-3}) - 9 f(x_{i-4}, y_{i-4}))$$

$$x = \begin{pmatrix} 0 \\ 0.1 \\ 0.2 \\ 0.3 \\ 0.4 \\ 0.5 \end{pmatrix} \qquad y = \begin{pmatrix} 0 \\ 0.0029 \\ 0.0131 \\ 0.0335 \\ 0.0677 \\ 0.1209 \end{pmatrix}$$

Результаты расчета:

$$y_4 = 0.0677 \qquad y_5 = 0.1209$$

Постановка задачи

Изучить одношаговые и многошаговые методы решения обыкновенных дифференциальных уравнений первого порядка с начальными условиями.

Используя метод Рунге - Кутта, вычислить значения y_1, y_2, y_3 приближенного решения заданного ДУ (согласно варианту заданий). Шаг сетки h для всех вариантов равен 0,1. Следующие два значения y_4, y_5 найти с помощью метода Адамса.

Содержание отчета

Отчет о лабораторной работе должен содержать:

- постановку задачи;
- пользовательскую функцию Mathcad, реализующую вычислительную схему метода Рунге - Кутта четвертого порядка;

- расчет трех значений y_1, y_2, y_3 приближенного решения задачи Коши по методу Рунге - Кутты;
- результаты вычисления в пакете Mathcad по формуле Адамса значений y_4, y_5 приближенного решения заданного ДУ.

Контрольные вопросы

1. Сформулируйте задачу Коши. С помощью каких численных методов можно ее решить?
2. Что является решением дифференциального уравнения?
3. Назовите источники погрешностей численного решения ДУ.
4. Какие общие черты присущи всем одношаговым методам?
5. В чем различие между одношаговыми и многошаговыми методами?
6. Что означает порядок метода?
7. Дайте графическую интерпретацию метода Эйлера.
8. С помощью графических построений объясните смысл каждого из коэффициентов K_0, K_1, K_2, K_3 формулы Рунге - Кутты.
9. В чем заключается идея метода Адамса? Почему этот метод называется экстраполяционным?

Варианты заданий

Номер варианта	Дифференциальное уравнение	Начальное условие
1	$y' = (x + y)^2 + 0,5(x + y)$	$y(0) = 0$
2	$y' = \frac{y}{1+x} - 0,5y^2$	$y(0) = 1$
3	$y' = 1 + (1 - x)y - (1,5 + x)y$	$y(0) = 2$
4	$y' = -(1 + xy) + 0,7(x^2 + y)$	$y(0) = 0$
5	$y' = (xy - 0,5)^2 + \frac{0,2}{1+x}$	$y(1) = 0$
6	$y' = (1 - y^2)x + 0,6y$	$y(0) = 2$
7	$y' = 1 - (x + y)^2 - \frac{2x^2}{2+x}$	$y(1) = 0$
8	$y' = (0,6 - y)x + 0,2y$	$y(0) = 0$
9	$y' = 0,2x^2 + 0,3(x^3 + y)$	$y(0) = 0$

Окончание вариантов заданий

10	$y' = (1,5x + y)^2 + x - y$	$y(0) = 1$
11	$y' = (xy - 0,4)^2 + \frac{0,3}{1 + y^2}$	$y(2) = 0$
12	$y' = 0,3(x - y)^2 + 0,5(x + y)$	$y(0) = 0$
13	$y' = 0,2(x - y)^3 + 0,3(x + y)$	$y(0) = 0$
14	$y' = 0,2(x - y) + (x + 0,1y)^2$	$y(0) = 1$
15	$y' = (0,6 - y^2)x + 0,2y$	$y(0) = 0$
16	$y' = 1 + (1,25 + x)y + y^2$	$y(0) = 1$
17	$y' = 1 + (1,25 - x)y - (1,75 + y)x$	$y(0) = 0$
18	$y' = x^2 + y^2 - 5xy$	$y(0) = 0$
19	$y' = x(x + 2y) - y^2$	$y(0) = 1$
20	$y' = 0,5(x^2 - y) + 1,5y^2$	$y(0) = 1$
21	$y' = (0,7 - x)y^2 + x^2$	$y(0) = 2$
22	$y' = (x^2 + 0,1y^2)x - 0,4$	$y(0) = 0$
23	$y' = x(3x + y) - y^2$	$y(0) = 0$
24	$y' = (x^2 - y^2)x + 1,5$	$y(1) = 0$
25	$y' = (x + x^2y)y + x^2$	$y(0) = 1$
26	$y' = (x - 0,3y)x^2 + \frac{1}{x^2 + 4}$	$y(0) = 1$
27	$y' = \frac{y}{1 + x} - y^2(x + 0,1)$	$y(0) = 1$
28	$y' = \frac{1,4}{x^2 + y^2 + 2}$	$y(1) = 0$
29	$y' = \frac{1}{y + 3x^2} + y + 1$	$y(1) = 0$
30	$y' = y(x - 1) - (2 - 0,5x)x$	$y(0) = 1$

Лабораторная работа № 7

МЕТОДЫ РЕШЕНИЯ ОБЫКНОВЕННЫХ ДИФФЕРЕНЦИАЛЬНЫХ УРАВНЕНИЙ. КРАЕВАЯ ЗАДАЧА

Цель работы:

- ознакомиться с конечно-разностными методами решения дифференциальных уравнений с граничными условиями;
- реализовать алгоритм метода прогонки для решения системы линейных конечно-разностных уравнений в пакете Mathcad;
- вычислить значения приближенного решения заданной краевой задачи.

Теоретические сведения

Краевая задача заключается в решении обыкновенного дифференциального уравнения при дополнительных условиях, поставленных при нескольких значениях независимой переменной, например:

$$\frac{d^2 y}{dx^2} = f(x, y, y') \quad (7.1)$$

при граничных условиях

$$y(a) = A, y(b) = B.$$

Численные методы решения таких задач разделяют на две группы:

- методы «стрельбы», основанные на замене решения краевой задачи решением нескольких задач Коши с последующим поиском таких начальных значений, чтобы выполнялись граничные условия;
- методы, в которых используют конечно-разностную форму дифференциального уравнения.

Достоинство **конечно-разностных методов** – они позволяют свести решение краевой задачи к решению системы алгебраических уравнений. Перейдем от непрерывной задачи к дискретной. При решении двухточечной краевой задачи (7.1) интервал $[a, b]$ можно разделить на n равных частей:

$$x_i = x_0 + ih, i = 1, \dots, n,$$

где $x_0 = a$, $x_n = b$, $h = \frac{b-a}{n}$. В узлах x_i требуется получить значения решения y_i . Заменяв производные y' и y'' по формулам численного дифференцирования на сетке с шагом h :

$$y'(x_i) \approx \frac{1}{2h}(y_{i+1} - y_{i-1}); \quad (7.2)$$

$$y''(x_i) \approx \frac{1}{h^2}(y_{i+1} - 2y_i + y_{i-1}), \quad (7.3)$$

можно представить ДУ в виде разностного уравнения. Отметим, что кроме (7.2) и (7.3) существуют и другие формы конечно-разностных выражений для производных, например:

$$y'(x_i) \approx \frac{1}{h}(y_{i+1} - y_i) \text{ или } y'(x_i) \approx \frac{1}{h}(y_i - y_{i-1}).$$

Если записать разностное уравнение для $i = 0 \dots n$ при двух граничных условиях, то получим систему из $(n - 1)$ алгебраических уравнений с $(n - 1)$ неизвестными. В результате решения этой системы получим таблицу приближенных значений искомой функции.

Пусть получена система линейных конечно-разностных уравнений. Типичная матрица коэффициентов такой системы трехдиагональная, т.е.

$$A = \begin{bmatrix} a_{11} & a_{12} & 0 & 0 & 0 & 0 \\ a_{21} & a_{22} & a_{23} & 0 & 0 & 0 \\ 0 & a_{32} & a_{33} & a_{34} & 0 & 0 \\ 0 & 0 & a_{43} & a_{44} & a_{45} & 0 \\ \vdots & \dots & \dots & \dots & \dots & \vdots \\ 0 & 0 & 0 & 0 & a_{n,n-1} & a_{nn} \end{bmatrix}.$$

Эта особенность матрицы коэффициентов системы уравнений в краевой задаче позволяет решать ее методом прогонки.

Метод прогонки – модифицированный вариант метода Гаусса для решения систем линейных уравнений с трехдиагональной матрицей коэффициентов.

Рассмотрим краевую задачу для линейного ДУ второго порядка

$$y'' + p(x)y' + q(x)y = f(x) \quad (7.4)$$

на интервале $[a, b]$ с условиями на краях интервала

$$\alpha_0 y(a) + \alpha_1 y'(a) = A, \quad (7.5)$$

$$\beta_0 y(b) + \beta_1 y'(b) = B.$$

Разобьем отрезок $[a, b]$ на n равных частей точками x_i с шагом h :

$$x_i = a + ih, \quad h = \frac{b-a}{n}, \quad i = 1, \dots, n.$$

Заменим производные искомой функции $y(x)$ в узлах x_i конечно-разностными выражениями (7.2), (7.3). Получим линейную систему

$$\begin{cases} \frac{y_{i+1} - 2y_i + y_{i-1}}{h^2} + p_i \frac{y_{i+1} - y_{i-1}}{2h} + q_i y_i = f_i, i = 1, \dots, n-1; \\ \alpha_0 y_0 + \alpha_1 \frac{y_1 - y_0}{h} = A; \\ \beta_0 y_n + \beta_1 \frac{y_n - y_{n-1}}{h} = B. \end{cases} \quad (7.6)$$

Разрешая каждое из $(n - 1)$ первых уравнений относительно y_i , получаем

$$y_i = \frac{2h^2 f_i}{2h^2 q_i - 4} - \frac{2 + p_i h}{2h^2 q_i - 4} y_{i+1} - \frac{2 - p_i h}{2h^2 q_i - 4} y_{i-1}. \quad (7.7)$$

Введем обозначения

$$s_i = \frac{2h^2 f_i}{2h^2 q_i - 4}; \quad m_i = \frac{2 + p_i h}{2h^2 q_i - 4}; \quad n_i = \frac{2 - p_i h}{2h^2 q_i - 4}, \quad i = 1, \dots, n-1, \quad (7.8)$$

тогда

$$y_i = s_i - m_i y_{i+1} - n_i y_{i-1}. \quad (7.9)$$

В результате систему (7.6) запишем в виде

$$\begin{cases} \alpha_0 y_0 + \alpha_1 \frac{y_1 - y_0}{h} = A; \\ y_i = s_i - m_i y_{i+1} - n_i y_{i-1} \quad (i = 1, \dots, n-1); \\ \beta_0 y_n + \beta_1 \frac{y_n - y_{n-1}}{h} = B. \end{cases} \quad (7.10)$$

Из первого уравнения системы (7.10) можно y_0 выразить через y_1 :

$$y_0 = c_0(d_0 - y_1).$$

Подставив y_0 во второе уравнение системы (7.10)

$$y_1 = s_1 - m_1 y_2 - n_1 y_0,$$

получим

$$y_1 = c_1(d_1 - y_2).$$

Продолжая этот процесс, приводим систему (7.10) к виду

$$\begin{cases} y_0 = c_0(d_0 - y_1); \\ y_1 = c_1(d_1 - y_2); \\ \vdots \\ y_{n-1} = c_{n-1}(d_{n-1} - y_n); \\ \beta_0 y_n + \beta_1 \frac{y_n - y_{n-1}}{h} = B. \end{cases} \quad (7.11)$$

Из первого уравнения системы (7.10) находим c_0 и d_0 :

$$y_0 = \frac{Ah}{\alpha_0 h - \alpha_1} - \frac{\alpha_1}{\alpha_0 h - \alpha_1} y_1 = \frac{\alpha_1}{\alpha_0 h - \alpha_1} \left(\frac{Ah}{\alpha_1} - y_1 \right) = c_0(d_0 - y_1),$$

следовательно,

$$c_0 = \frac{\alpha_1}{\alpha_0 h - \alpha_1}, \quad d_0 = \frac{Ah}{\alpha_1}. \quad (7.12)$$

Все последующие значения c_i и d_i ($i = 1, \dots, n - 1$) определим по формулам

$$c_i = \frac{m_i}{1 - n_i c_{i-1}}, \quad d_i = \frac{s_i - n_i c_{i-1} d_{i-1}}{m_i}. \quad (7.13)$$

Если $\alpha_1 = 0$, то выражения (7.12) неприменимы. В этом случае первое граничное условие имеет вид $y(a) = y_0 = A$. Тогда из второго уравнения системы (7.10) находим c_1 и d_1 :

$$y_1 = s_1 - m_1 y_2 - n_1 A = m_1 \left(\frac{s_1 - n_1 A}{m_1} - y_2 \right) = c_1(d_1 - y_2);$$

$$c_1 = m_1; \quad d_1 = \frac{s_1 - n_1 A}{m_1}. \quad (7.14)$$

Все последующие значения c_i и d_i ($i = 2, \dots, n - 1$) рассчитываем по формулам (7.13).

Из последних двух уравнений системы (7.11) находим y_n :

$$y_n = \frac{Bh + \beta_1 c_{n-1} d_{n-1}}{\beta_0 h + \beta_1 (1 + c_{n-1})}. \quad (7.15)$$

Если $\alpha_1 = 0$, $\beta_1 = 0$ (что соответствует задаче с закрепленными концами), то прогонка для разностной схемы устойчива. В этом случае ошибки, возникающие в ходе вычислений, не накапливаются, т.е. не зависят от n . Если решается краевая задача общего вида и $\alpha_1 \neq 0$, $\beta_1 \neq 0$, то при достаточно малых h прогонка устойчива, если

$$\alpha_0/\alpha_1 < 0, \quad \beta_0/\beta_1 > 0.$$

Таким образом, метод прогонки состоит из следующих этапов.

1. *Подготовительный.*

Вычисляем

$$x_i = a + ih, \quad i = 1, \dots, n;$$

$$p_i = p(x_i), \quad q_i = q(x_i), \quad f_i = f(x_i).$$

Затем по формулам (7.8) рассчитываем m_i , n_i , s_i .

2. *Прямой ход.*

Вычисляем c_0 и d_0 по выражениям (7.12), если $\alpha_1 \neq 0$, или c_1 и d_1 – по (7.14), если $\alpha_1 = 0$.

Все последующие значения c_i и d_i находим по формулам (7.13).

3. *Обратный ход.*

Определяем y_n по уравнению (7.15). Затем находим

$$\begin{aligned} y_{n-1} &= c_{n-1}(d_{n-1} - y_n); \\ y_{n-2} &= c_{n-2}(d_{n-2} - y_{n-1}); \\ &\vdots \\ y_1 &= c_1(d_1 - y_2); \\ y_0 &= \frac{Ah - \alpha_1 y_1}{\alpha_0 h - \alpha_1}. \end{aligned}$$

Отметим, что разностная схема аппроксимирует краевую задачу с точностью порядка h^2 .

Пример. Используя метод прогонки, найти значения (с шагом сетки $h = 0,1$) приближенного решения обыкновенного дифференциального уравнения второго порядка

$$y'' + xy' + (1 - x)y = 2 + 3x^2 - x^3,$$

удовлетворяющего граничным условиям

$$y(0) = 0, \quad y(0,5) = 0,25.$$

Зададим исходные данные в пакете Mathcad:

$$\begin{array}{lll} N := 5 & i := 0..N & h := 0.1 \\ A := 0 & B := 0.25 & y_0 := A \quad y_N := B \end{array}$$

Подготовительный этап:

$$x_i := h \cdot i \quad p_i := x_i \quad q_i := 1 - x_i \quad f_i := 2 + 3 \cdot (x_i)^2 - (x_i)^3$$

$$s_i := \frac{2 \cdot h^2 \cdot f_i}{2 \cdot h^2 \cdot q_i - 4} \quad m_i := \frac{2 + p_i \cdot h}{2 \cdot h^2 \cdot q_i - 4} \quad n_i := \frac{2 - p_i \cdot h}{2 \cdot h^2 \cdot q_i - 4}$$

Прямой ход (расчет коэффициентов c_i и d_i):

$$c_1 := m_1 \quad d_1 := \frac{s_1 - n_1 \cdot A}{m_1}$$

$$j := 2..N - 1$$

$$c_j := \frac{m_j}{1 - n_j \cdot c_{j-1}} \quad d_j := \frac{s_j - n_j \cdot c_{j-1} \cdot d_{j-1}}{m_j}$$

$$c = \begin{pmatrix} 0 \\ -0.5048 \\ -0.6768 \\ -0.7653 \\ -0.8199 \end{pmatrix} \quad d = \begin{pmatrix} 0 \\ 0.0202 \\ 0.0309 \\ 0.0424 \\ 0.0549 \end{pmatrix}$$

Обратный ход (нахождение значений y_i приближенного решения заданной краевой задачи на сетке $[0; 0,5]$ с шагом $h = 0,1$):

$$i := 1 .. 5$$

$$y_{N-i} := c_{N-i} \cdot (d_{N-i} - y_{N-i+1})$$

$$y = \begin{pmatrix} 0 \\ 0.01 \\ 0.04 \\ 0.09 \\ 0.16 \\ 0.25 \end{pmatrix}$$

Постановка задачи

Изучить метод конечных разностей для решения обыкновенных дифференциальных уравнений с граничными условиями и метод прогонки для трехдиагональной системы линейных уравнений.

Привести заданное (согласно варианту заданий) линейное дифференциальное уравнение второго порядка с двухточечными граничными условиями к системе конечно-разностных уравнений. Шаг сетки h для всех вариантов равен $0,1$.

Решить полученную систему методом прогонки.

Содержание отчета

Отчет о лабораторной работе должен содержать:

- постановку задачи;
- разностную схему заданной краевой задачи в виде системы линейных алгебраических уравнений (указать используемые при этом формулы);
- приближенные значения решения заданного ДУ с граничными условиями, вычисленные в пакете Mathcad с использованием метода прогонки.

Контрольные вопросы

1. Сформулируйте краевую задачу. Как с помощью численных методов можно ее решить?
2. Составьте несколько вариантов краевой задачи с различными граничными условиями.
3. От чего зависит точность решения краевой задачи?
4. Дайте характеристику метода прогонки: для чего он предназначен, в чем достоинство и недостатки, какое решение (точное или приближенное) позволяет получить.
5. Каким образом одно дифференциальное уравнение можно заменить системой линейных алгебраических уравнений? Всегда ли эта замена осуществима?
6. Выведите формулы для конечно-разностной аппроксимации производных первого, второго, третьего порядков.

Варианты заданий

Номер варианта	Дифференциальное уравнение	Граничные условия
1	$y'' + (x+3)y' - (x^2 + 2)y = 6x - x^2 - x^4$	$y(0) = 0;$ $y(0,5) = 1,25$
2	$y'' + xy' + (x-3)y = 6x + x^4$	$y(1) = 1,$ $y(1,5) = 3,37$
3	$y'' + (x+1)y' + x^2 y = 2x + x^4$	$y(0) = 1,$ $y(0,5) = 2$
4	$y'' + x^2 y' + (x^4 + 3)y = x + x^2$	$y(2) = 0,$ $y(2,5) = 1$
5	$y'' - (3-x)y' + (x^2 + 1)y = x^3 + 1$	$y(-0,5) = 1,$ $y(0) = 0,2$
6	$y'' - x^2 y' + (4 + x^3 + x^2)y = 3x + 5x^2$	$y(-1) = 0,$ $y(-0,5) = 1$
7	$y'' + xy' + x^3 y = 1 + 2x + 3x^2$	$y(0) = 1,$ $y(0,5) = 1,8$
8	$y'' + (x^2 + 4)y' - x^2 y = 1 + 4x^3$	$y(0) = 1,$ $y(0,5) = 3$
9	$y'' + (x+3)y' - (x^2 + 2)y = 6x - x^2 - x^4$	$y(-1,5) = 3,25,$ $y(-1) = 2$
10	$y'' - (x+1)y' + (x^3 + 4)y = x^2 + 1$	$y(-0,5) = 1,$ $y(0) = 2,5$
11	$y'' + (x+5)y' + 3x^2 y = 3x - x^3$	$y(1) = 0,$ $y(1,5) = 2,4$

Окончание вариантов заданий

Номер варианта	Дифференциальное уравнение	Граничные условия
12	$y'' - x^2 y' + (3 + x + x^2)y = x + x^2$	$y(2) = 1,$ $y(2,5) = -1,5$
13	$y'' + (x + x^2)y' + y = x^4 + 2x^3$	$y(0) = 0,$ $y(0,5) = 2$
14	$y'' + xy' + (x - 3)y = 6x + x^4$	$y(0) = 0,$ $y(0,5) = 0,42$
15	$y'' + (x - 4)y' + (x^2 + 5)y = 1 + x^4$	$y(1) = 1,$ $y(1,5) = 3,2$
16	$y'' + (x^3 + 1)y' + x^5 y = x^4$	$y(0) = 0,$ $y(0,5) = 1$
17	$y'' + xy' + (1 + 4x + x^2)y = x + x^2$	$y(-1) = 1,$ $y(-0,5) = 0,5$
18	$y'' - (3 + x)y' + (x^2 + 10)y = 1 - x^3$	$y(1) = 0,$ $y(1,5) = 2,4$
19	$y'' + (2x + 1)y' + 3x^2 y = x^4$	$y(2) = 1,$ $y(2,5) = 3,4$
20	$y'' - xy' + (5x^2 + 1)y = x + x^3$	$y(0) = 0,$ $y(0,5) = 2$
21	$y'' + (3x + 1)y' + x^3 y = 6x + x^2$	$y(1) = 0,$ $y(1,5) = 1,8$
22	$y'' - (x^2 + 1)y' + (x^3 + x)y = x^2$	$y(2) = 1,$ $y(2,5) = 3,4$
23	$y'' + xy' - (x^2 + 7)y = x + x^3$	$y(0) = 0,$ $y(0,5) = 1,4$
24	$y'' - (1 - 5x)y' + (x^2 + 1)y = 6 + x^3$	$y(1) = 3,$ $y(1,5) = 4,2$
25	$y'' + xy' + x^2 y = 1 + 3x + x^3$	$y(0) = 2,$ $y(0,5) = 3,5$
26	$y'' - (6x + 1)y' + (1 - x^3)y = 1 + x^4$	$y(0) = 0,$ $y(0,5) = 1,5$
27	$y'' + 3x^2 y' - (1 + x)y = x^3$	$y(1) = 2,$ $y(1,5) = -1$
28	$y'' - (x + 1)y' + (x^2 + 1)y = x + x^2$	$y(1,5) = 3,$ $y(2) = 1$
29	$y'' + 2xy' - (3x + 2)y = x^2 - x$	$y(2) = 1,$ $y(2,5) = 0$
30	$y'' - (2x^2 - 1)y' + 2xy = x^2$	$y(-1,5) = 0,$ $y(-1) = 1$

Лабораторная работа № 8

РЕШЕНИЕ ДИФФЕРЕНЦИАЛЬНЫХ УРАВНЕНИЙ В ЧАСТНЫХ ПРОИЗВОДНЫХ. МЕТОД СЕТОК

Цель работы:

- ознакомиться с методом сеток для решения различных типов дифференциальных уравнений в частных производных;
- реализовать алгоритмы решения ДУ эллиптического, параболического, гиперболического типов в пакете Mathcad;
- вычислить значения приближенных решений заданных ДУ.

Теоретические сведения

В инженерной практике часто приходится сталкиваться с задачами, в которых искомая величина – функция, зависящая от нескольких переменных (например, температура $u(x, t)$ зависит от координаты x и времени t). В этом случае решаемые уравнения содержат частные производные и называются ДУ в частных производных.

Для упрощения записи будем использовать такие обозначения:

$$u_t \equiv \frac{\partial u}{\partial t}, \quad u'_x \equiv \frac{\partial u}{\partial x}, \quad u''_{xx} \equiv \frac{\partial^2 u}{\partial x^2} \text{ и т.п.}$$

С математической точки зрения ДУ второго порядка в частных производных с двумя независимыми переменными

$$A(x, y)f''_{xx} + B(x, y)f''_{yy} + C(x, y)f''_{xy} + E(x, y, f, f'_x, f'_y) = 0 \quad (8.1)$$

классифицируются в зависимости от характера функций A , B и C , зависящих от переменных x и y . Если $B^2 - 4AC < 0$, то уравнение называется *эллиптическим*, если $B^2 - 4AC = 0$ – *параболическим*, а если $B^2 - 4AC > 0$ – *гиперболическим*. Например:

$$u'_t = u''_{xx}, \quad B^2 - 4AC = 0 \text{ (параболическое);}$$

$$u''_{tt} = u''_{xx}, \quad B^2 - 4AC = 4 \text{ (гиперболическое);}$$

$$u''_{xx} + u''_{yy} = 0, \quad B^2 - 4AC = -4 \text{ (эллиптическое).}$$

Зависимость функций A , B , C от x и y усложняет ситуацию, так как делает возможным изменение типа уравнения при переходе из одной части рассматриваемой области значений в другую. Дополнительными условиями для ДУ в частных производных могут служить граничные или начальные условия, а также комбинация тех и других.

Эллиптические уравнения описывают установившиеся (стационарные) процессы; задача ставится в замкнутой области и в каждой точке границы области задают граничные условия (задача Дирихле). Уравнения **параболического типа** описывают процессы теплопроводности и диффузии, а ДУ **гиперболического типа** – колебательные системы и волновые движения. Параболическими и гиперболическими уравнениями описывают эволюционные процессы (процессы распространения). В таких задачах на одной части границы ставятся граничные условия, на другой – начальные; возможны также открытые области, в которые «распространяется решение».

Некоторые ДУ в частных производных приведены в табл. 8.1.

Таблица 8.1

Известные типы ДУ в частных производных

Уравнение	Математическая форма	Примеры задач
Лапласа	$\Delta f = 0$	Установившееся течение жидкости. Стационарные тепловые поля
Пуассона	$\Delta f = -k$	Теплопередача с внутренними источниками тепла
Диффузии	$\Delta f = f'_t / h^2$	Нестационарная теплопроводность
Волновое	$\Delta f = f''_{tt} / c^2$	Распространение звуковых волн
Бигармоническое	$\Delta^2 f = F(x, y)$	Деформация пластин

Приняты следующие обозначения:

Δ – оператор Лапласа:

$\Delta f = f''_{xx}$ – в случае одной независимой переменной x ;

$\Delta f = f''_{xx} + f''_{yy}$ – для двух независимых переменных x и y ;

Δ^2 – бигармонический оператор, который в случае двух независимых переменных имеет вид

$$\Delta^2 f = \frac{\partial^4 f}{\partial x^4} + 2 \frac{\partial^4 f}{\partial x^2 \partial y^2} + \frac{\partial^4 f}{\partial y^4}.$$

В основе решения ДУ в частных производных методом сеток лежит конечно-разностная аппроксимация производных. Аппроксимация осуществляется в три этапа. Сначала в области решения вводят равномерную сетку узловых точек, соответствующую характеру задачи и граничным условиям. Затем решаемое уравнение записывают в наиболее удобной системе координат и, представляя производные в конечно-разностной форме, получают разностное уравнение, описы-

вающее функциональные связи между соседними узлами сетки. Разностное уравнение записывают для всех n узлов сетки. В результате получают систему n уравнений с n неизвестными. На последнем этапе систему n уравнений решают одним из численных методов.

Все приведенные уравнения в частных производных записаны в декартовой системе координат, однако иногда бывает удобнее пользоваться другими системами координат, обладающими специальными геометрическими свойствами и учитывающими физические особенности рассматриваемой задачи. Чаще всего применяют декартову, цилиндрическую и сферическую системы координат.

На рис. 8.1 показаны сетки, используемые для представления ДУ в частных производных в конечно-разностной форме. Нередко приходится иметь дело с областями неправильной формы. Границы такой области нельзя точно задать с помощью какой-либо одной из указанных выше сеток, существуют специальные методы, позволяющие модифицировать стандартные сетки так, что они позволят описать границу сложной конфигурации.

Применительно к эволюционным процессам сетку можно представить как последовательность узлов, простирающуюся в открытую область, насколько это необходимо для решения данной задачи.

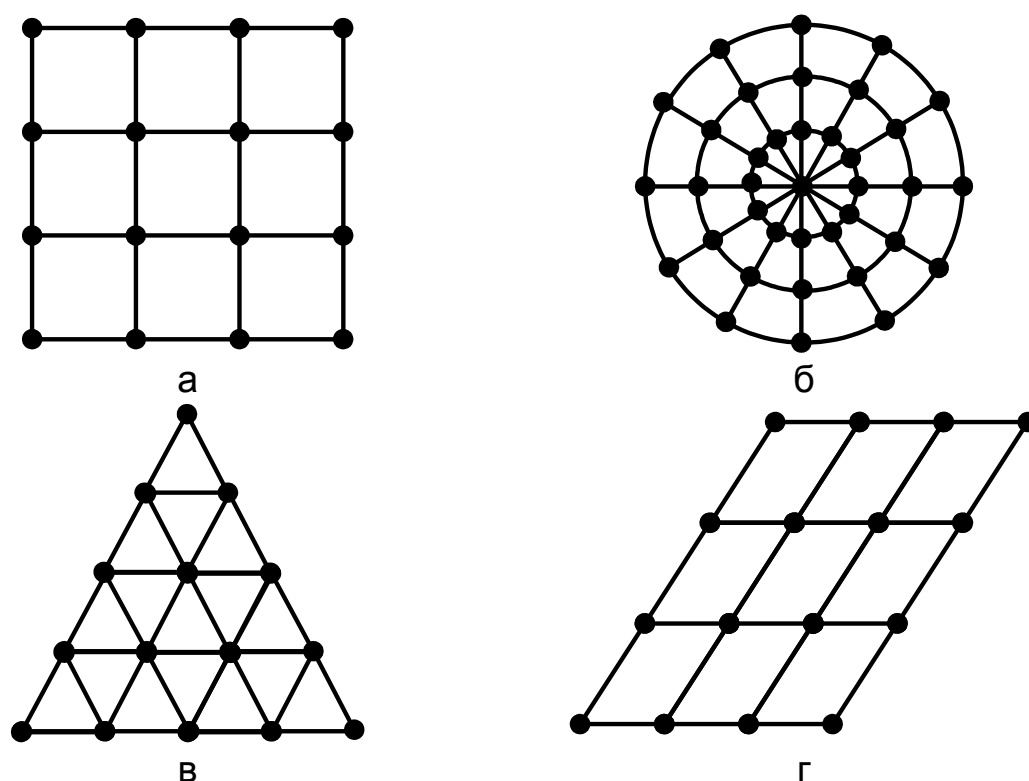


Рис. 8.1. Виды сеток, наиболее часто используемых при численном решении ДУ в частных производных: а – прямоугольная; б – полярная; в – треугольная; г – скошенная

Разложение в ряд Тейлора функции двух независимых переменных $f(x, y)$ позволяет представить частную производную в виде

$$f'_x \approx \frac{f(x_i + h, y_j) - f(x_i - h, y_j)}{2h}, \quad (8.1)$$

где h – малое приращение x относительно его значения в точке i .

В выражении (8.1) отброшены члены порядка h^2 , а само оно называется центрально-разностной формулой, так как симметрично относительно исходной точки (x_i, y_j) . Такое конечно-разностное представление удобно рассматривать как функциональную связь между тремя соседними узлами двумерной сетки с шагом h (рис. 8.2), где индекс j присвоен переменной y , а i относится к x .

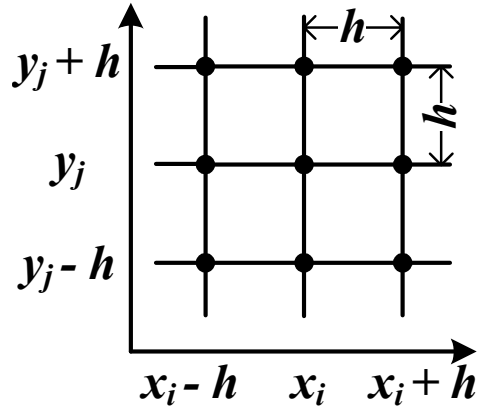


Рис. 8.2. Двумерная сетка

Заменим обозначение $f(x_i + h, y_j)$

на $f_{i+1,j}$. Тогда, используя разложение в ряд Тейлора, найдем выражения для частных производных

$$f'_x \approx \frac{f_{i+1,j} - f_{i-1,j}}{2h}; \quad (8.2)$$

$$f'_y \approx \frac{f_{i,j+1} - f_{i,j-1}}{2h}; \quad (8.3)$$

$$f''_{xx} \approx \frac{f_{i+1,j} - 2f_{i,j} + f_{i-1,j}}{h^2}; \quad (8.4)$$

$$f''_{yy} \approx \frac{f_{i,j+1} - 2f_{i,j} + f_{i,j-1}}{h^2}; \quad (8.5)$$

$$f''_{xy} \approx \frac{f_{i+1,j+1} - f_{i-1,j+1} - f_{i+1,j-1} + f_{i-1,j-1}}{4h^2}. \quad (8.6)$$

Применив разностную схему к каждому из n узлов сетки, получим систему n уравнений, которая будет линейной, если исходное дифференциальное уравнение имеет линейную структуру. Как правило, такие системы решают итерационными методами. Суть их заключается в многократном повторении одного и того же простого алгоритма, который дает результат, постепенно приближающийся к точному решению. Итерации начинаются с задания начального приближенного решения. Затем начальные значения переменных в узлах сетки

последовательно изменяются, пока не достигается заданная точность решения. Быстрота сходимости итерационного метода сильно зависит от степени точности начальной аппроксимации. Все приведенные разностные выражения (8.1) – (8.2) имеют погрешность порядка h^2 .

Метод сеток для уравнений эллиптического типа. Рассмотрим задачу нахождения стационарного распределения температуры в квадратной пластине, для которой заданы граничные условия:

$$u|_{\Gamma} = \varphi(x, y), \quad (8.7)$$

где $\varphi(x, y)$ – непрерывная функция, определяющая значение температуры на краях пластины. Распределение температуры в подобных случаях описывается уравнением Лапласа с двумя переменными

$$\Delta u = u''_{xx} + u''_{yy} = 0. \quad (8.8)$$

Математически требуется найти функцию $u(x, y)$, удовлетворяющую внутри области D уравнению (8.8), а на границе области Γ – условию (8.7). Для построения двумерной сетки проведем на плоскости XOY два семейства параллельных прямых

$$x_i = x_0 + ih \quad (i = 0, \pm 1, \pm 2, \dots); \quad y_j = y_0 + jl \quad (j = 0, \pm 1, \pm 2, \dots).$$

Точки пересечения этих прямых образуют узлы сетки. Рассмотрим узлы, которые принадлежат области $\bar{D} = D + \Gamma$. Узел (x_i, y_j) называют **внутренним**, если он и четыре соседних узла принадлежат области \bar{D} . Узлы области \bar{D} называют **граничными**, если хотя бы один соседний узел не принадлежит \bar{D} .

Проанализируем ДУ (8.8) во всех внутренних узлах:

$$u''_{xx}|_{(x_i, y_j)} + u''_{yy}|_{(x_i, y_j)} = 0. \quad (8.9)$$

Заменим частные производные второго порядка разностными отношениями. В результате получим приближенное равенство

$$\frac{u_{i+1, j} - 2u_{i, j} + u_{i-1, j}}{h^2} + \frac{u_{i, j+1} - 2u_{i, j} + u_{i, j-1}}{l^2} = 0. \quad (8.10)$$

Наиболее простой вид уравнение (8.10) имеет при $h = l$:

$$u_{i, j} = \frac{1}{4}(u_{i+1, j} + u_{i-1, j} + u_{i, j+1} + u_{i, j-1}). \quad (8.11)$$

Рассмотрим граничные узлы. Если узел (x_i, y_j) попадает на границу Γ области D , то значение $u_{i, j}$ можно вычислить с помощью условия $u|_{\Gamma} = \varphi(x, y)$. Если узел (x_i, y_j) не попадает на границу Γ , то значение $u_{i, j}$ полагаем равным значению функции $\varphi(x, y)$ в точке

границы Γ , ближайшей к этому узлу. Иногда значение $u_{i,j}$ полагают равным значению $\varphi(x,y)$ в точке M границы Γ , ближайшей к рассматриваемому граничному узлу в направлении оси Ox или Oy :

$$u_{i,j} \approx \varphi(M). \quad (8.12)$$

Объединив (8.11) и (8.12), получим линейную систему, в которой число уравнений равно числу неизвестных. Эта система всегда совместна и имеет единственное решение. Если число узлов сетки велико, то для решения системы удобно применять итерационный метод. Выбираем начальные приближения $u_{i,j}^{(0)}$ для всех внутренних узлов сетки. Их можно получить путем линейной интерполяции по известным граничным значениям. Например, если в граничных узлах (x_0, y_j) и (x_n, y_j) значения функции $u_{0,j}$ и $u_{n,j}$ найдены из условия (8.7), то значения во внутренних узлах $u_{i,j}^{(0)}$, $i = 1, \dots, n - 1$ рассчитываем по формуле

$$u_{i,j}^{(0)} = u_{0,j} + \frac{u_{n,j} - u_{0,j}}{n} i.$$

Следующие приближения для внутренних узлов вычисляем по (8.11):

$$u_{i,j}^{(k+1)} = \frac{1}{4} (u_{i+1,j}^{(k)} + u_{i-1,j}^{(k)} + u_{i,j+1}^{(k)} + u_{i,j-1}^{(k)}).$$

Итерации продолжают до тех пор, пока в двух последовательных приближениях не совпадет требуемое количество десятичных знаков.

Функция, реализующая метод сеток для ДУ эллиптического типа:

```

Net1( $\phi$ ,  $K$ ) :=
|
|  $r \leftarrow rows(\phi)$ 
|
|  $c \leftarrow cols(\phi)$ 
|
| for  $k \in 0 .. K$ 
|
|   for  $j \in 1 .. c - 2$ 
|
|     for  $i \in 1 .. r - 2$ 
|
|        $\phi_{i,j} \leftarrow \frac{(\phi_{i+1,j} + \phi_{i-1,j} + \phi_{i,j+1} + \phi_{i,j-1})}{4}$ 
|
|
|  $\phi$ 

```

Аргументы функции: ϕ – имя матрицы значений функции в граничных узлах сетки и начальных значений функции во внутренних узлах сетки; K – число итераций метода.

Функция, реализующая итерационную процедуру поиска решения ДУ заданного типа и возвращающая решение с точностью ε :

$$\begin{array}{l}
 \mathit{DESolve}(\phi_0, \mathit{Net}, \varepsilon) := \left\{ \begin{array}{l}
 K \leftarrow 1 \\
 \phi_1 \leftarrow \mathit{Net}(\phi_0, K) \\
 \Delta \leftarrow \max\left(\overrightarrow{|\phi_1 - \phi_0|}\right) \\
 \mathit{while} \ \Delta > \varepsilon \\
 \quad \left\{ \begin{array}{l}
 K \leftarrow K + 1 \\
 \phi_0 \leftarrow \phi_1 \\
 \phi_1 \leftarrow \mathit{Net}(\phi_0, K) \\
 \Delta \leftarrow \max\left(\overrightarrow{|\phi_1 - \phi_0|}\right)
 \end{array} \right. \\
 \phi_1
 \end{array} \right.
 \end{array}$$

Аргументы функции: ϕ_0 – имя матрицы начальных значений решения в узлах сетки; Net – имя функции, реализующей вычислительную схему для решения ДУ; ε – точность вычислений.

Пример. Найти решение уравнения Лапласа в квадрате с вершинами $A(0, 0)$, $B(0, 1)$, $C(1, 1)$, $D(1, 0)$, если $u|_{AB} = 4y$, $u|_{BC} = 4$, $u|_{CD} = 2y(1 + y)$, $u|_{AD} = 4x(1 - x)$. Шаг сетки $h = 1/4$.

Зададим исходные данные в пакете Mathcad:

$$i := 0..4 \qquad j := 0..4 \qquad x_i := \frac{i}{4} \qquad y_i := \frac{i}{4}$$

Найдем значения решения в граничных узлах, начальные значения в девяти внутренних узлах полагаем равными нулю:

$$\begin{array}{l}
 U_{i,0} := 4 \cdot y_i \\
 U_{i,4} := 2 \cdot y_i \cdot (1 + y_i) \\
 U_{0,j} := 4 \cdot x_j \cdot (1 - x_j) \\
 U_{4,j} := 4
 \end{array}
 \qquad
 U = \begin{pmatrix}
 0 & 0.75 & 1 & 0.75 & 0 \\
 1 & 0 & 0 & 0 & 0.625 \\
 2 & 0 & 0 & 0 & 1.5 \\
 3 & 0 & 0 & 0 & 2.625 \\
 4 & 4 & 4 & 4 & 4
 \end{pmatrix}$$

Получим решение ДУ эллиптического типа с точностью $\varepsilon = 0,01$:

$$W := \mathit{DESolve}(U, \mathit{Net1}, 0.01)$$

$$W = \begin{pmatrix} 0 & 0.75 & 1 & 0.75 & 0 \\ 1 & 1.316 & 1.399 & 1.173 & 0.625 \\ 2 & 2.114 & 2.109 & 1.917 & 1.5 \\ 3 & 3.03 & 3.007 & 2.887 & 2.625 \\ 4 & 4 & 4 & 4 & 4 \end{pmatrix}$$

Метод сеток для уравнений параболического типа. Рассмотрим решение смешанной краевой задачи для уравнения

$$u'_t = u''_{xx}, \quad 0 \leq x \leq b, \quad t \geq 0 \quad (8.13)$$

с граничными условиями

$$u|_{x=0} = \varphi(t), \quad u|_{x=b} = \psi(t)$$

и начальным условием

$$u|_{t=0} = f(x).$$

Это уравнение описывает распределение температуры в бесконечно тонком стержне. Для решения задачи построим прямоугольную сетку, узлы которой (x_i, t_j) определяются следующими формулами:

$$x_i = ih \quad (i = 1, \dots, n), \quad \text{где } h = b/n \text{ – шаг вдоль оси } Ox;$$

$$t_j = jl \quad (j = 0, 1, 2, \dots), \quad \text{где } l \text{ – шаг вдоль оси } Ot.$$

Рассмотрим ДУ (8.13) во внутренних узлах:

$$u'_t|_{(x_i, t_j)} = u''_{xx}|_{(x_i, t_j)}.$$

Заменим частные производные конечными разностями:

$$\frac{u_{i,j+1} - u_{i,j}}{l} = \frac{u_{i+1,j} - 2u_{i,j} + u_{i-1,j}}{h^2}. \quad (8.14)$$

Обозначим $l/h^2 = \sigma$. Разрешая (8.14) относительно $u_{i,j+1}$, получаем

$$u_{i,j+1} = (1 - 2\sigma)u_{i,j} + \sigma(u_{i+1,j} + u_{i-1,j}). \quad (8.15)$$

Таким образом, значения функции u в $(j+1)$ -м слое определяются ее значениями на j -м слое.

Значения функции на нулевом слое ($j = 0$) находим из начального условия

$$u(x_i, 0) = u_{i,0} = f(x_i) = f_i, \quad i = 1, \dots, n.$$

Значения функции u в граничных узлах, лежащих на прямых $x = 0$ и $x = b$, можно определить из граничных условий

$$u(0, t_j) = u_{0,j} = \varphi(t_j) = \varphi_j; \quad u(b, t_j) = u_{n,j} = \psi(t_j) = \psi_j, \quad j = 1, 2, \dots$$

Схема (8.15) устойчива при $\sigma \leq 1/2$. Наименьшая погрешность замены дифференциального уравнения конечно-разностным выражением при $\sigma = 1/6$. В этом случае формула (8.15) примет вид

$$u_{i,j+1} = \frac{1}{6}(u_{i-1,j} + 4u_{i,j} + u_{i+1,j}). \quad (8.16)$$

Полученное конечно-разностное выражение справедливо для всех внутренних узлов и позволяет явным образом выразить значение функции u в момент времени t_{j+1} через значение функции u в предыдущий момент времени t_j . Такая постановка задачи позволяет обойтись без системы совместных уравнений и, следовательно, не требует применения итерационных методов.

Функция, реализующая метод сеток для ДУ параболического типа:

```

Net2( $\phi$ ) :=
|
|  $r \leftarrow rows(\phi)$ 
|  $c \leftarrow cols(\phi)$ 
| for  $j \in 0..c-2$ 
|   for  $i \in 1..r-2$ 
|      $\phi_{i,j+1} \leftarrow \phi_{i,j} + \sigma \cdot (\phi_{i+1,j} - 2 \cdot \phi_{i,j} + \phi_{i-1,j})$ 
|  $\phi$ 

```

Аргумент функции: ϕ – имя матрицы начальных значений решения в узлах сетки.

Пример. С помощью метода сеток найти решение ДУ $u'_t = u''_{xx}$, удовлетворяющее условиям $u|_{t=0} = (1,5x^2 + 1,4)\sin(\pi x)$, $0 \leq x \leq 1$; $u|_{x=0} = \varphi(t) = 32t(1+t)$, $u|_{x=1} = \psi(t) = 0$ для значений $0 \leq t \leq 1/24$, взяв по аргументу x шаг $h = 1/4$, а по аргументу $t - l = h^2/6 = 1/96$.

Зададим исходные данные в пакете Mathcad:

$$\begin{array}{l}
i := 0..4 \\
j := 0..4
\end{array}
\quad
\Delta t := \frac{1}{96}
\quad
\Delta x := \frac{1}{4}
\quad
\sigma := \frac{\Delta t}{\Delta x^2}
\quad
\begin{array}{l}
x_i := \Delta x \cdot i \\
t_j := \Delta t \cdot j
\end{array}$$

$$\begin{array}{l}
f(x, a, c) := (a \cdot x^2 + c) \cdot \sin(\pi \cdot x) \\
\phi(t) := 32 \cdot t \cdot (1 + t) \\
u_{i,0} := f(x_i, a, c)
\end{array}
\quad
\begin{array}{l}
a := 1.5 \quad c := 1.4 \\
\psi(t) := 0 \\
u_{0,j} := \phi(t_j) \quad u_{4,j} := \psi(t_j)
\end{array}$$

Получим решение ДУ эллиптического типа:

$$u := \text{Net2}(u) \quad u = \begin{pmatrix} 0 & 0.337 & 0.681 & 1.031 & 1.389 \\ 1.056 & 1 & 0.993 & 1.022 & 1.077 \\ 1.775 & 1.624 & 1.475 & 1.344 & 1.238 \\ 1.587 & 1.354 & 1.173 & 1.028 & 0.909 \\ 0 & 0 & 0 & 0 & 0 \end{pmatrix}$$

Метод сеток для уравнений гиперболического типа. Наиболее часто встречающийся в инженерной практике вид гиперболического ДУ в частных производных – волновое уравнение, описывающее различные виды колебаний (колебания струны или мембраны, распространение звуковых волн в жидкости и т.п.).

Рассмотрим уравнение

$$u''_{tt} = u''_{xx}, \quad 0 \leq x \leq b, \quad t \geq 0. \quad (8.17)$$

Чтобы полностью поставить задачу, выражение (8.17) необходимо дополнить двумя граничными и двумя начальными условиями:

$$\begin{aligned} u|_{x=0} &= \varphi(t), \quad u|_{x=b} = \psi(t), \\ u|_{t=0} &= f(x), \quad u'_t|_{t=0} = F(x). \end{aligned}$$

Решаем задачу методом сеток. Строим в полуполосе $t \geq 0$ и $0 \leq x \leq b$ два семейства параллельных прямых:

$$\begin{aligned} x_i &= ih \quad (i = 1, \dots, n), \quad \text{где } h = b/n \text{ – шаг вдоль оси } 0x; \\ t_j &= jl \quad (j = 0, 1, 2, \dots), \quad \text{где } l \text{ – шаг вдоль оси } 0t. \end{aligned}$$

Узлы, лежащие на прямых $x = 0$, $x = b$ и $t = 0$, – граничные, остальные – внутренние.

Рассмотрим ДУ (8.17) во внутренних узлах. Заменяем частные производные центральными конечными разностями:

$$u_{i,j+1} = 2(1 - \sigma^2)u_{i,j} + \sigma^2(u_{i+1,j} + u_{i-1,j}) - u_{i,j-1}, \quad (8.18)$$

где $\sigma = l/h$.

Из формулы (8.18) видно, что для получения значений $u(x, t)$ в $(j + 1)$ -м слое необходимы значения $u(x, t)$ в двух предыдущих слоях.

Значения функции $u(x, t)$ на нулевом слое ($j = 0$) задаются первым начальным условием

$$u_{i,0} = f(x_i) = f_i, \quad i = 1, \dots, n.$$

Значения $u(x, t)$ на первом слое ($j = 1$) можно приближенно найти из второго начального условия, заменив $u'_t \approx \frac{u_{i,1} - u_{i,0}}{l}$:

$$u_{i,1} = u_{i,0} + lF_i, i = 1, \dots, n - 1.$$

Значения функции $u(x, t)$ на прямых $x = 0$ и $x = b$ можно вычислить, используя граничные условия

$$u_{0,j} = \varphi(t_j) = \varphi_j, u_{n,j} = \psi(t_j) = \psi_j, j = 1, 2, \dots$$

Доказано, что рассмотренная разностная схема устойчива при $\sigma \leq 1$. При $\sigma = 1$, т.е. $l = h$, расчетная формула (8.18) примет вид

$$u_{i,j+1} = u_{i+1,j} + u_{i-1,j} - u_{i,j-1}. \quad (8.19)$$

Как и в случае параболического уравнения, это выражение допускает смещение вперед по времени, не прибегая к итерациям. Однако в отличие от предыдущего последнее рекуррентное соотношение позволяет получить следующее значение функции u при $t = t_{j+1}$ при условии, что значения функции u известны как при $t = t_j$, так и при $t = t_{j-1}$.

Функция, реализующая метод сеток для ДУ гиперболического типа:

$$Net3(\phi) := \left| \begin{array}{l} r \leftarrow rows(\phi) \\ c \leftarrow cols(\phi) \\ \text{for } j \in 1..c-2 \\ \quad \text{for } i \in 1..r-2 \\ \quad \quad \phi_{i,j+1} \leftarrow \phi_{i+1,j} - \phi_{i,j-1} + \phi_{i-1,j} \\ \phi \end{array} \right.$$

Аргумент функции такой же, как и для $Net2(\phi)$.

Пример. Найти решение ДУ $u''_{tt} = u''_{xx}$, удовлетворяющее условиям $u|_{t=0} = f(x) = (1,8x + 0,6)\sin(\pi x)$, $u'_t|_{t=0} = F(x) = 0$, $0 \leq x \leq 1$; $u|_{x=0} = \varphi(t) = 0$, $u|_{x=1} = \psi(t) = 0$ для значений $0 \leq t \leq 0,8$ на сетке с шагом $h = l = 0,2$.

Зададим исходные данные в пакете Mathcad:

$$\begin{array}{llll} \Delta t := 0.2 & \Delta x := 0.2 & i := 0..5 & j := 0..4 \\ x_i := \Delta x \cdot i & t_j := \Delta t \cdot j & \phi(t) := 0 & \psi(t) := 0 \\ f(x) := (1.8 \cdot x + 0.6) \cdot \sin(\pi \cdot x) & & & F(x) := 0 \end{array}$$

С помощью начальных и граничных условий найдем значения решения в граничных узлах:

$$\begin{aligned}
 u_{i,0} &:= f(x_i) & u_{0,j} &:= \phi(t_j) \\
 u_{i,1} &:= u_{i,0} + \Delta t \cdot F(x_i) & u_{5,j} &:= \psi(t_j)
 \end{aligned}$$

Получим решение ДУ гиперболического типа:

$$u := \text{Net3}(u)$$

$$u = \begin{pmatrix} 0 & 0 & 0 & 0 & 0 \\ 0.564 & 0.564 & 0.691 & 0.342 & -0.399 \\ 1.255 & 1.255 & 0.907 & 0.292 & -0.857 \\ 1.598 & 1.598 & 0.857 & -0.292 & -0.907 \\ 1.199 & 1.199 & 0.399 & -0.342 & -0.691 \\ 0 & 0 & 0 & 0 & 0 \end{pmatrix}$$

Постановка задачи

Изучить разностные схемы метода сеток для решения ДУ в частных производных эллиптического, параболического, гиперболического типов.

Применяя метод сеток с шагом $h = 1/4$, найти решение уравнения Лапласа в квадрате с вершинами $A(0, 0)$, $B(0, 1)$, $C(1, 1)$, $D(1, 0)$ при заданных (согласно варианту заданий А) краевых условиях.

Найти приближенное решение ДУ $u'_t = u''_{xx}$, удовлетворяющее условиям $u|_{t=0} = (ax^2 + c)\sin(\pi x)$, $0 \leq x \leq 1$, $u|_{x=0} = \phi(t)$, $u|_{x=1} = \psi(t)$ (согласно варианту заданий Б), для значений $0 \leq t \leq 1/24$, взяв по аргументу x шаг сетки $h = 1/4$, а по аргументу $t - l = 1/96$.

Решить ДУ $u''_{tt} = u''_{xx}$ при условиях $u|_{t=0} = f(x)$ (согласно варианту заданий В), $u'_t|_{t=0} = 0$, $0 \leq x \leq 1$; $u|_{x=0} = u|_{x=1} = 0$ для значений $0 \leq t \leq 0,8$ на сетке с шагом $h = l = 0,2$.

Содержание отчета

Отчет о лабораторной работе должен содержать:

- постановку задачи;
- разностные схемы решения поставленных задач;
- вычисление в пакете Mathcad методом сеток приближенных решений заданных ДУ в частных производных.

Контрольные вопросы

1. Сформулируйте физическую и математическую постановку задач, требующих решения ДУ в частных производных.
2. В чем заключается идея метода сеток? Какие сетки используют при численном решении ДУ в частных производных?
3. Чем определяется точность решения ДУ в частных производных? Как оценить точность решения уравнения Лапласа?
4. Что означает устойчивость решения?
5. С чем связан выбор соотношения сторон сеток?
6. Почему для решения ДУ эллиптического типа используют итерационную процедуру, а для решения ДУ параболического и гиперболического типов итерация не требуется?

Варианты заданий А

Номер варианта	$u _{AB}$	$u _{BC}$	$u _{CD}$	$u _{AD}$
1	$20y(1-y^2)$	0	0	$20\sin(\pi x)$
2	$30\sin(\pi y)$	$20x$	$20y$	$30x(1-x)$
3	$40y^2$	$40(1-x^2)$	$40y^2(1-y)$	0
4	$30y$	$30(1-x^2)$	0	0
5	$25\sin(2\pi y)$	$25x(1-x)$	$25\sin[0,5\pi(1-y)]$	$25\sin(0,5\pi x)$
6	$40y$	$10(x-2)^2$	$10y^2$	$10\sin(\pi x)$
7	$25y^3$	$25(1-x^2)$	$20(1-y)$	$30x$
8	$10(1-y)$	$20x^2$	20	$10(1+x)$
9	$30y$	$30\cos(0,5\pi x)$	$30\cos(0,5\pi y)$	$x(29+x)$
10	$15y(1-y^2)$	$10\sin(\pi x)$	$20\sin(\pi y)$	0
11	$15y$	15	$15y^2$	$20x(1-x)$
12	$30\sqrt{y}$	30	$30y^2$	$10\sin(\pi x)$
13	$10y$	10	$5(1+y^2)$	$5x$
14	$20y$	$20(1-x)^3$	$15(1-y)$	$15\sin(0,5\pi x)$
15	0	$30x(1-x)$	$30y(1-y^2)$	$30x(1-x)$
16	$40y^2$	$40(1-x)$	0	$30\sin(2\pi x)$
17	$25(1-y)$	$30x^2$	$5(5+y)$	25
18	$30(1-y)$	$20\sqrt{x}$	$20y$	$30(1-x)$
19	$25\sqrt{y}$	$25(1-x)$	0	$10x(1-x)$
20	0	$20x^2(1-x)$	$30(1-y^2)$	$30x$

Варианты заданий Б

Номер варианта	a	c	$\varphi(t)$	$\psi(t)$
1	1,1	1,2	$16t$	$4t(t+2)$
2	1,3	1,4	$8t(t+1)$	$16t$
3	1,4	1,2	$32t$	$24t(2t+1)$
4	1,6	1,1	$0,5\sin(32\pi t)$	$16t$
5	1,2	1,1	$32(1-t)$	0
6	1,7	1,2	$0,7\sin(24\pi t)$	$32t$
7	1,5	1,1	$32t(1-2t)$	$16t(t+1)$
8	1,0	1,4	$0,5\sin(24\pi t)$	$8t$
9	1,8	1,0	$8t(t+1)$	0
10	1,3	1,2	0	$0,4\sin(32\pi t)$
11	1,2	1,4	$4t(2t+1)$	$4t(1-2t)$
12	1,1	1,3	0	$32t(t+2)$
13	1,4	1,3	$16t(t+1)$	$8t$
14	1,4	1,7	$16t$	$\sin(24\pi t)$
15	1,7	1,3	$0,6\sin(32\pi t)$	0
16	1,5	1,2	0	$8t(t+1)$
17	1,0	1,6	$32t(1-t)$	$16t$
18	1,6	1,3	$8t$	$16t(t+1)$
19	1,1	1,4	$32t$	$16t(4t+1)$
20	1,0	1,5	$0,3\sin(24\pi t)$	0

Варианты заданий В

Номер варианта	1	2	3	4
$f(x)$	$0,5x(1-x)$	$1,1(1+x^2)\sin(\pi x)$	$0,4x^2(1-x)$	$0,6x\sin(\pi x)$
Номер варианта	5	6	7	8
$f(x)$	$0,9x^2(1-x)$	$(x+x^2)\sin(\pi x)$	$0,2x(1-x)$	$(1,2+1,1x^2)\sin(\pi x)$
Номер варианта	9	10	11	12
$f(x)$	$1,5x^3(1-x)$	$1,4x^2(1-x^2)$	$1,2x^2(1-x)$	$(x^2+0,5x)\sin(\pi x)$
Номер варианта	13	14	15	16
$f(x)$	$1,5x(1-x^2)$	$0,7x\sin(\pi x)$	$1,1x^3(1-x)$	$x(2+x)\sin(\pi x)$
Номер варианта	17	18	19	20
$f(x)$	$2,1x^2(1-x^2)$	$(1,8x+1)\sin(\pi x)$	$1,8(x-x^2)$	$(1,3x^2+1,1)\sin(\pi x)$

Библиографический список

Боглаев, Ю.П. Вычислительная математика и программирование [Текст] : учеб. пособие для студентов вузов / Ю.П. Боглаев. – М. : Высш. шк., 1990. – 544 с.

Михайленко, С.В. Численные методы [Текст] : учеб. пособие по курсу «Прикладная и вычислительная математика» / С.В. Михайленко. – Х. : Харьк. авиац. ин-т, 1978. – 127 с.

Михайленко, С.В. Численные методы [Текст] : учеб. пособие по лаб. практикуму / С.В. Михайленко, Л.И. Черноштан. – Х. : Харьк. авиац. ин-т, 1980. – 126 с.

Половко, А.М. Mathcad для студента [Текст] / А.М. Половко, И.В. Ганичев. – СПб. : БХВ-Петербург, 2006. – 336 с.

Поршнеv, С.В. Численные методы на базе Mathcad [Текст] / С.В. Поршнеv, И.В. Беленкова. – СПб. : БХВ-Петербург, 2005. – 464 с.

Шуп, Т. Решение инженерных задач на ЭВМ [Текст] : практ. руководство: пер. с англ. / Т. Шуп; под ред. В.Б. Миносцева. – М. : Мир, 1982. – 238 с.

Содержание

Введение.....	3
Лабораторная работа № 1. Численное решение алгебраических и трансцендентных уравнений	4
Лабораторная работа № 2. Итерационные методы решения систем алгебраических линейных уравнений	15
Лабораторная работа № 3. Методы обработки табличных данных. Интерполяция.....	27
Лабораторная работа № 4. Численное интегрирование.....	42
Лабораторная работа № 5. Аппроксимация табличных данных кривыми.....	47
Лабораторная работа № 6. Методы решения обыкновенных дифференциальных уравнений. Задача Коши.	57
Лабораторная работа № 7. Методы решения обыкновенных дифференциальных уравнений. Краевая задача.	66
Лабораторная работа № 8. Решение дифференциальных уравнений в частных производных. Метод сеток.....	74
Библиографический список.....	88

Навчальне видання

**Попов Анатолій Владиславович
Васильєва Ірина Карлівна**

**Обчислювальні методи в проектуванні
радіоелектронної апаратури**

(Російською мовою)

Редактор В.І.Філатова

Зв. план, 2011

Підписано до друку 14.04.2011

Ум. друк. арк. 5. Обл.-вид. арк. 5,62. Електронне видання

Національний аерокосмічний університет ім. М.Є. Жуковського
"Харківський авіаційний інститут"
61070, Харків-70, вул. Чкалова, 17
<http://www.khai.edu>

Видавничий центр "ХАІ"
61070, Харків-70, вул. Чкалова, 17
izdat@khai.edu

Свідоцтво про внесення суб'єкта видавничої справи до Державного реєстру видавців, виготовлювачів і розповсюджувачів видавничої продукції, серія ДК № 391, видане Державним комітетом інформаційної політики, телебачення та радіомовлення України від 30.03.2001 р.