

С.Д. Шапорев

**МЕТОДЫ ВЫЧИСЛИТЕЛЬНОЙ МАТЕМАТИКИ
И ИХ ПРИЛОЖЕНИЯ**

Учебное пособие

**Санкт-Петербург
2002**

Министерство образования Российской Федерации
Балтийский государственный технический университет «Военмех»

С.Д. Шапорев

МЕТОДЫ ВЫЧИСЛИТЕЛЬНОЙ МАТЕМАТИКИ
И ИХ ПРИЛОЖЕНИЯ

Учебное пособие

Санкт-Петербург
2002

УДК 681, 683

Ш 24

Шапорев С.Д. Методы вычислительной математики и их приложения: учебное пособие / Балт. гос. техн. ун-т «Военмех». СПб., 2002. 230 с.

В пособии рассматриваются основные вычислительные методы, наиболее часто используемые в практике инженерных расчетов по специальностям выпускающих кафедр БГТУ: методы теории приближения функций, численное дифференцирование и интегрирование, методы решения задач линейной алгебры и нелинейных систем уравнений, решение обыкновенных дифференциальных уравнений и систем дифференциальных уравнений, методы решения дифференциальных уравнений с частными производными.

Большое внимание уделяется практической работе с описанными алгоритмами, предлагаются лабораторные работы по всем изучаемым темам, написанные в математическом пакете Mathcad. Каждая лабораторная работа включает серию индивидуальных заданий.

Пособие предназначено для студентов дневного и вечернего отделения. Его использование поможет активизировать самостоятельную работу студентов по курсу «Вычислительная математика» и даст возможность преподавателям контролировать индивидуальную работу студентов в течение всего семестра.

Р е ц е н з е н т ы: кафедра высшей математики ПГУПС (зав. каф. доктор техн. наук, проф. *В.Г. Дегтярев*), доктор техн. наук, проф. *М.С. Попов*.

*Утверждено
редакционно-издательским
советом университета*

1. ЭЛЕМЕНТАРНАЯ ТЕОРИЯ ПОГРЕШНОСТЕЙ; ВЫЧИСЛИТЕЛЬНЫЕ ЗАДАЧИ, МЕТОДЫ И АЛГОРИТМЫ

Инженерные вычислительные задачи в настоящее время решаются с помощью ЭВМ и имеют некоторые характерные особенности:

- 1) ярко выраженная практическая направленность: необходимость доведения результатов до конкретных чисел;
- 2) большой объем вычислительных работ;
- 3) сложность математических моделей процессов;
- 4) широкое использование готовых вычислительных методов.

При решении задач пункт три имеет определяющее значение. Математическая модель – это приближенное описание исследуемого процесса на языке математики. Для любой задачи математическая модель никогда не бывает полностью адекватной. Сам процесс построения моделей всегда протекает итеративно, модель при исследовании непрерывно усложняется, происходит ее подгонка и адаптация, улучшение качества. Решение инженерной задачи с помощью ЭВМ обычно содержит следующие этапы:

- 1) постановка задачи;
- 2) выбор математической модели;
- 3) постановка вычислительной задачи;
- 4) анализ вычислительных свойств вычислительной задачи;
- 5) выбор или создание нужного численного метода;
- 6) алгоритмизация и программирование;
- 7) обработка и интерпретация результата;
- 8) использование результата и коррекция математической модели.

1.1. Источники и классификация погрешностей результата численного эксперимента

Точный результат решения получить численно невозможно, он всегда содержит погрешность, то есть является приближенным. Наличие погрешности решения обусловлено следующими причинами.

1. Математическая модель всегда приближена. Характеристики истинного процесса всегда отличаются от модельных характеристик.
2. Исходные данные всегда содержат погрешность, ибо результат эксперимента неизбежно получается с ошибкой.
3. Теоретические и численные методы решения задач являются приближенными. Лишь решение тривиальной задачи можно получить в виде конечной формулы.
4. При вводе и выводе с ЭВМ производятся округления. Такие же округления производятся и при вычислениях.

Если y - точное значение вычисляемой величины, то это значение содержит следующие погрешности: $\delta_{\text{н}}y$ - неустранимая погрешность (пункты 1 и 2), $\delta_{\text{м}}y$ - погрешность метода (пункт 3), $\delta_{\text{в}}y$ - вычислительная погрешность (пункт 4). Таким образом, $\delta y = y - y^* = \delta_{\text{н}}y + \delta_{\text{м}}y + \delta_{\text{в}}y$. Не следует думать, что $\delta_{\text{н}}y$ совершенно неизвестна. Конечно, она не известна точно, но ее можно оценить приближенно, адаптируя модель к изучаемому процессу $\delta_{\text{м}}y$ должна быть примерно на порядок меньше $\delta_{\text{н}}y$, а $\delta_{\text{в}}y$ на порядок меньше $\delta_{\text{м}}y$. В этом случае достигается разумный компромисс и повышается достоверность конечного результата.

1.2. Погрешности чисел

Пусть a - точное и неизвестное значение некоторой величины, а a^* - ее известное приближенное значение.

Ошибкой (или погрешностью) приближенного значения числа a^* называется разность $a - a^*$. Количество мерой ошибки является абсолютная погрешность

$$\Delta(a^*) = |a - a^*|. \quad (1.2.1)$$

По ней не всегда можно сделать правильное заключение о качестве приближения. Для этого вводится понятие относительной погрешности.

Относительной погрешностью приближенного значения числа a^* называется

$$\delta(a^*) = \frac{|a - a^*|}{|a|} = \frac{\Delta(a^*)}{|a|}. \quad (1.2.2)$$

Эта погрешность не зависит от масштаба величины единицы измерения. Непосредственное вычисление по формулам (1.2.1) и (1.2.2) невозможно, так как a неизвестно. Часто задают величины $\bar{\Delta}(a^*) \geq |a - a^*|$, $\bar{\delta}(a^*) \geq \frac{|a - a^*|}{|a|}$ – верхние границы погрешностей и полагают

$$\bar{\delta}(a^*) \approx \frac{\bar{\Delta}(a^*)}{|a^*|}, \quad \bar{\Delta}(a^*) \approx |a^*| \cdot \bar{\delta}(a^*) \quad (1.2.3)$$

При записи приближенных чисел руководствуются следующими правилами. Пусть a^* задано в виде конечной десятичной дроби $a^* = \alpha_n \alpha_{n-1} \dots \alpha_0 \beta_1 \beta_2 \dots \beta_m$.

Значащими цифрами числа a^* называются все цифры в его записи, начиная с первой ненулевой слева. Например,

$$\begin{cases} a_1^* = \underline{2.173}, \\ a_2^* = 0.0\underline{103}, \quad \text{все значащие цифры подчеркнуты.} \\ a_3^* = 0.01035600. \end{cases}$$

Значащую цифру числа a^* называют верной, если абсолютная погрешность числа не превосходит единицы разряда, соответствующего этой цифре.

Пусть $a^* = 0.032105$ и $\bar{\Delta}(a^*) = 2 \cdot 10^{-5}$. Тогда a^* имеет три верные значащие цифры (они подчеркнуты). Если бы $\bar{\Delta}(a^*) < 1 \cdot 10^{-5}$, было бы четыре верные значащие цифры. Число верных значащих цифр тесно связано с величиной относительной погрешности числа. Имеют место следующие результаты.

Теорема 1.1. Если a^* содержит N верных значащих цифр, то $\delta(a^*) \leq (10^{N-1} - 1)^{-1} \approx 10^{-N+1}$.

Теорема 1.2. Для того чтобы число a^* содержало N верных значащих цифр, достаточно, чтобы $\delta(a^*) \leq (10^N + 1)^{-1} \approx 10^{-N}$.

Теорема 1.3. Если a^* имеет ровно N значащих цифр, то $10^{-N-1} \leq \delta(a^*) \leq 10^{-N+1}$, то есть $\delta(a^*) \approx 10^{-N}$.

Эти теоремы позволяют оценивать $\delta(a^*)$ по числу значащих цифр и наоборот. Например, если дано $a^* = 0.364001$, то есть a^* имеет шесть значащих цифр, то $\delta(a^*) \approx 10^{-6}$.

При округлении возникает погрешность, называемая погрешностью округления. Существуют два способа округления.

1. Усечение – отбрасывание всех цифр, расположенных правее n -ой значащей цифры. При этом погрешность ε не превышает (достигает) единицы того же разряда.

2. Округление по дополнению. Это следующее правило: если первая цифра слева от отбрасываемых меньше пяти, то лишене просто отбрасывается, как при усечении; если же первая цифра слева от отбрасываемых больше или равна пяти, то в младший сохраняется

разряд добавляется единица. Абсолютная величина погрешности по дополнению не превышает половины единицы последней оставляемой значащей цифры.

1.3. Погрешности арифметических операций

Теорема 1.4. Абсолютная погрешность алгебраической суммы не превосходит суммы абсолютных погрешностей слагаемых, то есть $\Delta(a^* \pm b^*) \leq \Delta(a^*) + \Delta(b^*)$.

Доказательство

Запишем формулу для абсолютной погрешности алгебраической суммы двух величин по определению и воспользуемся свойствами модуля. Получим сразу необходимый результат

$$\Delta(a^* \pm b^*) = |(a \pm b) - (a^* \pm b^*)| = |(a - a^*) \pm (b - b^*)| \leq |a - a^*| + |b - b^*| = \Delta(a^*) + \Delta(b^*). \quad (1.3.1)$$

Теорема 1.5. Пусть a и b - ненулевые числа одного знака. Тогда

$$\begin{aligned} \delta(a^* + b^*) &\leq \delta_{\max}, \quad \delta(a^* - b^*) \leq 9 \cdot \delta_{\max}, \\ \delta_{\max} &= \max\{\delta(a^*), \delta(b^*)\}, \quad 9 = \frac{|a + b|}{|a - b|}. \end{aligned} \quad (1.3.2)$$

Доказательство

Поступим так же, как и в предыдущей теореме. Выразим абсолютную погрешность через относительную:

$$\begin{aligned} \Delta(a^* \pm b^*) &= |a \pm b| \delta(a^* \pm b^*) \leq \Delta(a^*) + \Delta(b^*) = |a| \delta(a^*) + |b| \delta(b^*) \leq (|a| + |b|) \delta_{\max} = |a + b| \delta_{\max}. \\ \delta(a^* + b^*) &= \frac{\Delta(a^* + b^*)}{|a + b|} = \frac{|a + b| \delta_{\max}}{|a + b|} = \delta_{\max}. \\ \delta(a^* - b^*) &= \frac{\Delta(a^* - b^*)}{|a - b|} = \frac{|a + b| \delta_{\max}}{|a - b|} = \frac{|a + b|}{|a - b|} \delta_{\max}. \end{aligned} \quad (1.3.3)$$

Из последних двух равенств видно, что при вычитании двух близких чисел может произойти катастрофическая потеря точности, так как при $|a - b| \rightarrow 0$ в последней формуле $\delta(a^* - b^*) \rightarrow \infty$.

Теорема 1.6. Для относительных погрешностей произведения и частного приближенных чисел верны следующие оценки:

$$\begin{aligned} \delta(a^* \cdot b^*) &\leq \delta(a^*) + \delta(b^*) + \delta(a^*) \cdot \delta(b^*), \\ \delta\left(\frac{a^*}{b^*}\right) &\leq \frac{\delta(a^*) + \delta(b^*)}{1 - \delta(b^*)}. \end{aligned} \quad (1.3.4)$$

Доказательство

Первая требуемая формула получается традиционным путем:

$$\begin{aligned} |ab| \delta(a^* b^*) &= \Delta(a^* b^*) = |ab - a^* b^*| = |(a - a^*)b + (b - b^*)a - (a - a^*)(b - b^*)| \leq \\ &\leq |b|\Delta(a^*) + |a|\Delta(b^*) + \Delta(a^*)\Delta(b^*) = |a| \cdot |b| \left\{ \frac{|b|\Delta(a^*)}{|a| \cdot |b|} + \frac{|a|\Delta(b^*)}{|a| \cdot |b|} + \frac{\Delta(a^*)}{|a|} \cdot \frac{\Delta(b^*)}{|b|} \right\} = \\ &= |a| \cdot |b| [\delta(a^*) + \delta(b^*) + \delta(a^*) \cdot \delta(b^*)]. \end{aligned}$$

Для вывода второй оценки предварительно получим, используя свойства модуля, следующую формулу: $|b^*| = |b + (b^* - b)| \geq |b| - \Delta(b^*) = |b| \cdot [1 - \delta(b^*)]$. Тогда

$$\delta\left(\frac{a^*}{b^*}\right) = \frac{\left|\frac{a}{b} - \frac{a^*}{b^*}\right|}{\left|\frac{a}{b}\right|} = \frac{|ab^* - ba^*|}{|ab^*|} = \frac{|a(b^* - b) + b(a - a^*)|}{|ab^*|} \leq \frac{|a|\Delta(b^*) + |b|\Delta(a^*)}{|ab^*|} = \frac{\delta(a^*) + \delta(b^*)}{1 - \delta(b^*)}.$$

Следствие. Если $\delta(b^*) << 1$, а $\delta(a^*) \leq 1$, то для оценки границ относительных погрешностей можно использовать следующие приближенные равенства $\bar{\delta}(a^*b^*) \approx \bar{\delta}\left(\frac{a^*}{b^*}\right) \approx \bar{\delta}(a^*) + \bar{\delta}(b^*)$. Чаще всего на практике делают именно так.

1.4. Погрешности функций

Пусть $f(\bar{x}) = f(x_1, x_2, \dots, x_n)$ - функция n -переменных, дифференцируемая в рассматриваемой области (например, на отрезке $[x, x^*]$).

Теорема 1.7. Для абсолютной погрешности значения $y^* = f(\bar{x}^*)$ справедлива следующая формула: $\Delta(y^*) \leq \sum_{j=1}^n \max_{[x, x^*]} |f'_{x_j}| \cdot \Delta(x_j)$. (1.4.1)

Доказательство

Вспомним сначала формулу Тейлора* для функции нескольких переменных. Для функции одного переменного разложение в окрестности точки x^* будет иметь вид

$$f(x) = f(x^*) + \frac{f'(x^*)}{1!}(x - x^*) + \frac{f''(x^*)}{2!}(x - x^*)^2 + \dots + \frac{f^{(n-1)}(x^*)}{(n-1)!}(x - x^*)^{n-1} + R_n(\theta), \theta \in (x, x^*)$$

Для функции n переменных форма записи формулы Тейлора остается точно такой же, если вместо производных записать дифференциалы соответствующих порядков:

$$\Delta f(\bar{x}) = \frac{1}{1!} df(\bar{x}^*) + \frac{1}{2!} d^2 f(\bar{x}^*) + \dots + \frac{1}{(n-1)!} d^{n-1} f(\bar{x}^*) + R_n(\bar{\theta}).$$

где, $d^n f(\bar{x}) = \left(\frac{\partial}{\partial x_1} dx_1 + \frac{\partial}{\partial x_2} dx_2 + \dots + \frac{\partial}{\partial x_n} dx_n \right)^n f(\bar{x})$. Например, для функции двух переменных $f(x_1, x_2) = f(x_1^*, x_2^*) + \sum_{k=1}^{n-1} \frac{1}{k!} \left[(x_1 - x_1^*) \frac{\partial}{\partial x_1} + (x_2 - x_2^*) \frac{\partial}{\partial x_2} \right]^k f(x_1^*, x_2^*) + R_n(\bar{\theta}) =$

$$= f(x_1^*, x_2^*) + \frac{1}{1!} \frac{\partial}{\partial x_1} f(x_1^*, x_2^*)(x_1 - x_1^*) + \frac{1}{1!} \frac{\partial}{\partial x_2} f(x_1^*, x_2^*)(x_2 - x_2^*) + \frac{1}{2!} \frac{\partial^2}{\partial x_1^2} f(x_1^*, x_2^*)(x_1 - x_1^*)^2 + \\ + \frac{1}{2!} \frac{\partial^2}{\partial x_1 \partial x_2} f(x_1^*, x_2^*)(x_1 - x_1^*)(x_2 - x_2^*) + \frac{1}{2!} \frac{\partial^2}{\partial x_2^2} f(x_1^*, x_2^*)(x_2 - x_2^*)^2 + \dots$$

Отбрасывая все члены второго порядка и выше, получим

$$f(x_1, x_2) - f(x_1^*, x_2^*) = \Delta(y^*) \approx f'_{x_1}(x_1^*, x_2^*)(x_1 - x_1^*) + f'_{x_2}(x_1^*, x_2^*)(x_2 - x_2^*)$$

Таким образом, искомая формула сразу вытекает из формулы Лагранжа**. Если $\bar{x} - \bar{x}^*$ достаточно мало, то для предельных значений погрешностей можно положить

$$\bar{\Delta}(y^*) = \sum_{j=1}^n \left| f'_{x_j}(\bar{x}^*) \right| \bar{\Delta}(x_j^*) \text{ или } \bar{\Delta}(y^*) = \sum_{j=1}^n \left| f'_{x_j}(\bar{x}) \right| \bar{\Delta}(x_j^*).$$

Для относительных погрешностей тогда имеем следующие формулы:

* Брук Тейлор (1685-1731) - английский математик.

** Жозеф Луи де Лагранж (1736-1813) - французский математик, механик и астроном.

$$\begin{aligned}\bar{\delta}(y^*) &= \frac{\bar{\Delta}(y^*)}{|y|} = \frac{\bar{\Delta}(y^*)}{\left|f\left(\bar{x}^*\right)\right|} = \frac{\sum_{j=1}^n \left|f'_{x_j}\left(\bar{x}^*\right)\right| \bar{\Delta}(x_j^*) \cdot |x_j^*|}{\left|f\left(\bar{x}^*\right)\right| \cdot |x_j^*|} = \frac{\sum_{j=1}^n |x_j^*| f'_{x_j}\left(\bar{x}^*\right)}{\left|f\left(\bar{x}^*\right)\right|} \cdot \frac{\bar{\Delta}(x_j^*)}{|x_j^*|} = \sum_{j=1}^n \vartheta_j^* \bar{\delta}(x_j^*) \text{ или} \\ \bar{\delta}(y^*) &= \sum_{j=1}^n \vartheta_j \bar{\delta}(x_j^*), \text{ где } \vartheta_j = \frac{|x_j^*| f'_{x_j}(x)}{\|f(x)\|}.\end{aligned}\quad (1.4.2)$$

I частный случай. Функция $y = f(x)$ - функция одного переменного. Здесь следует положить $n = 1$, тогда $\bar{\Delta}(y^*) \approx |f'(x^*)| \bar{\Delta}(x^*)$ или $\bar{\Delta}(y^*) \approx |f'(x)| \bar{\Delta}(x^*)$. Для относительных погрешностей все аналогично:

$$\vartheta^* = \frac{|x^*| f'(x^*)}{\|f(x^*)\|}, \bar{\delta}(y^*) = \vartheta^* \bar{\delta}(x^*) \text{ или } \vartheta = \frac{|x| f'(x)}{\|f(x)\|}, \bar{\delta}(y^*) = \vartheta \bar{\delta}(x^*). \quad (1.4.3)$$

II частный случай. Функция $F(\bar{x}, y) = 0$ - неявная. Этот случай отличается от исходного только формулой для нахождения частных производных:

$$f'_{x_j}(\bar{x}) = \left. \frac{-F'_{x_j}(\bar{x}, y)}{F'_y(\bar{x}, y)} \right|_{y=f(\bar{x}), j=1, 2, \dots, n}.$$

Пример. Пусть корни квадратного уравнения $x^2 + bx + c = 0$ вычисляются при коэффициентах $b \approx 10^3$, $c \approx 1$. Каково влияние погрешностей задания коэффициентов на точность вычисляемых значений?

Решение

Договоримся сразу, чтобы избежать путаницы, об обозначениях. Здесь роль двух функций ($y_{1,2}$ в формулах 1.4.1 и 1.4.2) играют x_1 и x_2 - корни квадратного уравнения. Роль переменных x_1, x_2, \dots, x_n играют b и c - коэффициенты квадратного уравнения. Их всего два. В этих обозначениях $x_1 \equiv y_1(b, c) = f_1(b, c)$, $x_2 \equiv y_2(b, c) = f_2(b, c)$. Однако, этих явных уравнений нет, они лишь предполагаются, задана же неявная функция $x^2 + bx + c = F(\bar{x}, b, c) = 0$.

Найдем выражения для частных производных $F'_x = 2x + b$, $F'_b = x$, $F'_c = 1$.

Отсюда $|f'_b| = \left| \frac{F'_b}{F'_x} \right|$, $|f'_c| = \left| \frac{F'_c}{F'_x} \right|$. Тогда по формуле (1.4.2) получим

$$\begin{aligned}\bar{\delta}(x_1^*) &= \vartheta_{1,1} \bar{\delta}(b^*) + \vartheta_{2,1} \bar{\delta}(c^*), \bar{\delta}(x_2^*) = \vartheta_{1,2} \bar{\delta}(b^*) + \vartheta_{2,2} \bar{\delta}(c^*), \text{ где} \\ \vartheta_{1,1} &= \frac{|b^*| f'_b(b^*, c^*)}{|x_1|}, \vartheta_{2,1} = \frac{|c^*| f'_c(b^*, c^*)}{|x_1|}, \vartheta_{1,2} = \frac{|b^*| f'_b(b^*, c^*)}{|x_2|}, \vartheta_{2,2} = \frac{|c^*| f'_c(b^*, c^*)}{|x_2|}.\end{aligned}$$

Здесь первый индекс у коэффициента ϑ - номер переменного, а второй индекс соответствует номеру функции, то есть номеру корня квадратного уравнения. Решим исходное квадратное уравнение. $x^2 + bx + c = 0 \Rightarrow x_1 \approx -10^3$, $x_2 \approx -10^{-3}$. Тогда

$$\begin{aligned}\vartheta_{1,1} &= \frac{|b^*| \left| \frac{F'_b}{F'_x} \right|}{|x_1|} \Bigg|_{x=x_1, b^*=10^3, c^*=1} \approx \frac{10^3 \left| \frac{-10^3}{-2 \cdot 10^3 + 10^3} \right|}{|-10^3|} = 1. \text{ Аналогично, } \vartheta_{2,1} = 10^{-6}, \vartheta_{1,2} = 1, \\ \vartheta_{2,2} &= 1. \text{ Итак, } \begin{cases} \bar{\delta}(x_1^*) = \bar{\delta}(b^*) + 10^{-6} \cdot \bar{\delta}(c^*), \\ \bar{\delta}(x_2^*) = \bar{\delta}(b^*) + \bar{\delta}(c^*) \end{cases}\end{aligned}$$

1.5. Особенности машинной арифметики

Причиной появления вычислительных погрешностей является способ представления чисел на ЭВМ. В привычной нам десятичной системе счисления вещественное число x представляется последовательностью символов, начиная со знака \pm , и продолжается цепочкой десятичных цифр α_i и β_i , то есть

$$x = \pm\alpha_n\alpha_{n-1}...\alpha_1\alpha_0.\beta_1\beta_2...\beta_m...$$

Так как десятичная система счисления позиционная, то значение числа вычисляется так:

$$x = \pm(\alpha_n \cdot 10^n + \alpha_{n-1} \cdot 10^{n-1} + \dots + \alpha_1 \cdot 10^1 + \alpha_0 \cdot 10^0 + \beta_1 \cdot 10^{-1} + \beta_2 \cdot 10^{-2} + \dots + \beta_m \cdot 10^{-m} + \dots).$$

Все вычислительные машины работают в двоичной системе счисления. В ней то же число выглядит следующим образом:

$$x = \pm(\alpha_n \cdot 2^n + \alpha_{n-1} \cdot 2^{n-1} + \dots + \alpha_1 \cdot 2^1 + \alpha_0 \cdot 2^0 + \beta_1 \cdot 2^{-1} + \beta_2 \cdot 2^{-2} + \dots + \beta_m \cdot 2^{-m} + \dots).$$

По форме представления, способу хранения и реализации вычислительных операций на ЭВМ целые и вещественные числа существенно различаются.

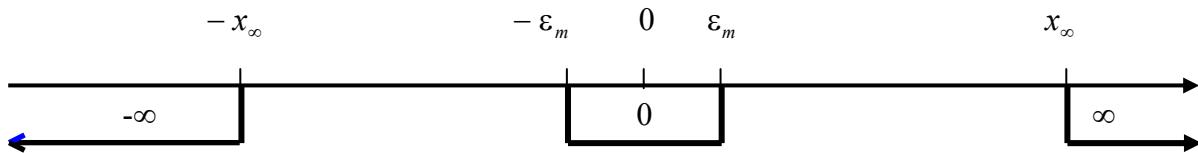
Целые представляются так: $n = \pm(\alpha_L \cdot 2^L + \dots + \alpha_1 \cdot 2^1 + \alpha_0 \cdot 2^0)$, где L - фиксированное число. Всего для хранения числа n отводят $L+2$ разряда, в том числе один разряд на знак числа. Таким образом, $n_{\max} = 2^L + \dots + 2^0 = 2^{L+1} - 1$. L обычно не слишком велико, например, для ЭВМ РС $L = 30$. Тогда $n_{\max} = 2^{31} - 1 \approx 2 \cdot 10^9$.

Арифметические операции сложения, вычитания и умножения производятся точно, если результат меньше или равен n_{\max} . Если же это не так, то ситуация не доводится до сведения пользователя, а результату присваивается некоторое значение меньшее абсолютной величины n_{\max} .

Вещественные числа представляются в форме с плавающей точкой, то есть в виде $x = \pm(\gamma_1 \cdot 2^{-1} + \gamma_2 \cdot 2^{-2} + \dots + \gamma_t \cdot 2^{-t}) \cdot 2^p$, где обычно $\gamma_1 = 1$, t - размерность мантиссы, p - двоичный порядок. Порядок также записывается как двоичное целое число $p = \pm(\sigma_l, \sigma_{l-1}, \dots, \sigma_0)_2$, на его хранение отводятся $l+2$ двоичных разрядов $\underbrace{\pm \gamma_1 \gamma_2 \gamma_3 \dots \gamma_t}_{\text{мантисса}} \underbrace{\pm \sigma_l \sigma_{l-1} \sigma_{l-2} \dots \sigma_0}_{\text{порядок}}$. Следует помнить, что на ЭВМ представимы не все числа, а

лишь конечный набор рациональных чисел. Это – представимое множество данной вычислительной машины. Для всех остальных чисел возможно лишь их приближенное представление с ошибкой представления (ошибкой округления). Если округление производится усечением, то $\overline{\delta}(x^*) = \varepsilon_m = 2^{-t+1}$, если имеет место округление по дополнению, то $\varepsilon_m = 2^{-t}$, ε_m называется машинной точностью или машинным эпсилоном.

Диапазон изменения чисел в ЭВМ ограничен. Если $x > 2^{p_{\max}}$, то $x = x_{\infty}$, $p = 2^{l+1} - 1$. Если же $x < \varepsilon_m$, то $x = 0$.



Для ЭВМ РС диапазон представления вещественных чисел примерно равен: $10^{-38} < |x| < 3 \cdot 10^{38}$, при этом $\varepsilon_m \approx 10^{-7}$.

1.6. Лабораторная работа № 1. Определение абсолютной и относительной погрешностей приближенных чисел. Оценка погрешностей результата

При проведении расчетов необходимо уметь практически оценивать погрешности ре-

зультата. Разумная оценка погрешности позволяет удерживать оптимальное число знаков при вычислениях, оптимизируя трудоемкость расчетов. Абсолютная и относительная погрешности вычисляются непосредственно по формулам (1.2.1) и (1.2.2). Число верных знаков определяется в соответствии с теоремами 1-3 параграфа 1.2.

Пример. Округляя заданное число a до трех значащих цифр, определить абсолютную Δ и относительную δ погрешности полученных приближенных чисел.

Если, например, $a = 2.1514$, то $a^* = 2.15$ и $\Delta(a^*) = 0.0014$. Тогда $\delta(a^*) = \frac{\Delta(a^*)}{|a|} = \frac{0.0014}{2.1514} = 0.00065 = 0.07\%$. Аналогично, если $a = -0.0015281$, то $a^* = -0.00153$, $\Delta(a^*) = |a - a^*| = 0.0000019$, $\delta(a^*) = 0.00124 = 0.12\%$. Здесь при округлении числа a использовано обычное правило округления по дополнению.

Погрешности арифметических операций вычисляются по формулам параграфа 1.3. При этом, например, погрешность суммы большого числа слагаемых, вычисленная по формуле (1.3.1), оказывается сильно завышенной. Поэтому часто применяют статистическую оценку погрешности суммы:

$$\Delta(S) = \sqrt{3n} \cdot 0.5 \cdot 10^{-m}, \quad (1.6.1)$$

где n - число слагаемых в сумме S , то есть $S = a_1 + a_2 + \dots + a_n$, m - номер десятичного разряда, до которого произведено округление. Формулу (1.6.1) используют при $n > 10$.

Относительная погрешность разности двух положительных чисел, определяемая по формуле (1.3.3), может быть очень большой, особенно когда эти числа близки между собой. Чтобы избежать потери точности, в этом случае часто используют преобразования типа преобразований суммы в произведение.

Пример. Найти алгебраическую сумму приближенных чисел и указать ее погрешность: $S = a + b + c = 398.5 - 72.28 + 0.34567$ (все знаки верные).

Начнем со слагаемых a, b и c . Будем считать, что абсолютная погрешность каждого из них не превосходит половины единицы младшего оставленного разряда, то есть $\Delta(a^*) = 0.05$, $\Delta(b^*) = 0.005$, $\Delta(c^*) = 0.000005$. Тогда $\delta(a^*) = 0.00013 = 0.01\%$, $\delta(b^*) = 0.00007 = 0.01\%$, $\delta(c^*) = 0.00001 = 0.001\%$. По формуле (1.3.1) $\Delta(S^*) = \Delta(a^*) + \Delta(b^*) + \Delta(c^*) = 0.05$, то есть абсолютная погрешность суммы не превосходит абсолютной погрешности наименее точного из слагаемых. Сохраним в расчетах только один запасной знак и будем округлять все слагаемые до 0.01. Тогда $S = 398.5 - 72.28 + 0.35 = 326.57$. Окончательный результат округляем до 0.1. $S = 326.6$. При этом к вычисленной ранее абсолютной погрешности 0.05 добавляется погрешность округления, также равная 0.05. Таким образом, окончательно $\Delta(S^*) = 0.05 + 0.05 = 0.1$. Отсюда $\delta(S^*) = \frac{0.01}{326.6} = 0.0003 = 0.03\%$.

Для оценки относительных погрешностей результата при умножении и делении формулы (1.3.4) используются редко. Обычно на практике выполняются условия $\delta(a^*) \leq 1$, $\delta(b^*) \ll 1$ и используется приближенное равенство

$$\delta(a^* \cdot b^*) \approx \delta\left(\frac{a^*}{b^*}\right) \approx \delta(a^*) + \delta(b^*). \quad (1.6.2)$$

Таким образом, практически при умножении и делении приближенных чисел их относительные погрешности складываются.

Пример. Высота h и радиус основания R цилиндра измерены с точностью до 0.5% и оказались равными 2.06 и 6.5 см. Какова предельная относительная и абсолютная погрешности при вычислении объема цилиндра?

$V = S_{\text{осн}} \cdot h = \pi R^2 h$. Число π может быть записано с произвольно высокой точностью. Для R и h имеем $\delta(R^*) = 0.5\% = 0.005$, $\delta(h^*) = 0.5\% = 0.005$. $\delta(V^*) \approx \delta(R^*) + \delta(R^*) + \delta(h^*) = 2\delta(R^*) + \delta(h^*) = 0.015 = 1.5\%$. Таким образом, результат содержит не более двух верных знаков. При вычислении V сохраним один запасной знак, округляя все числа до трех знаков: $V = 3.142 \cdot 2.06 \cdot 6.5^2 = 273.464$. Тогда $V = 273.0$, $\Delta(V^*) = |V| \cdot \delta(V^*) = 4.0$.

Погрешности вычисления значений функций подсчитываются по формулам (1.4.1) и (1.4.2). Если значения функций положительны, то для относительной погрешности можно использовать формулу

$$\delta(y^*) = \sum_{i=1}^n \frac{1}{f} \left| \frac{\partial f}{\partial x_i} \right| \Delta(x_i^*) = \sum_{i=1}^n \left| \frac{\partial \ln f}{\partial x_i} \right| \Delta(x_i^*). \quad (1.6.3)$$

Для функции одной переменной легко можно определить допустимую погрешность аргумента по допустимой погрешности функции

$$\Delta(x^*) = \frac{1}{|f'(x)|} \Delta(y^*). \quad (1.6.4)$$

Для функции нескольких аргументов эта задача решается при введении дополнительного предположения – так называемого **принципа равных влияний**. При этом полагают, что в формуле (1.4.1) все слагаемые $|f'_{x_i}| \Delta(x_i^*)$ равны между собой, тогда

$$\Delta(x_i^*) = \frac{\Delta(y^*)}{n \left| \frac{\partial f}{\partial x_i} \right|}, \quad i = \overline{1, n}. \quad (1.6.5)$$

Пример. С каким числом верных знаков должен быть известен свободный член уравнения $x^2 - 2x + \lg 2 = 0$, чтобы получить корни с четырьмя верными знаками?

Будем считать, что коэффициент p в квадратном уравнении известен абсолютно точно. Для вычисления корней используем выражение $x_{1,2} = -\frac{p}{2} \pm \sqrt{\left(\frac{p}{2}\right)^2 - q}$.

По условию $\delta(x_1^*) = \delta(x_2^*) = 10^{-4} = 0.01\%$. По формуле (1.4.2)

$$\delta(x^*) = \left| \frac{\partial x}{\partial p} \right| \frac{|p^*|}{|x^*|} \delta(p^*) + \left| \frac{\partial x}{\partial q} \right| \frac{|q^*|}{|x^*|} \delta(q^*)$$

Если бы оба коэффициента влияли на точность корней, то можно было бы воспользоваться для вычисления $\delta(q^*)$ принципом равных влияний. Однако в нашем случае p известен точно, то есть $\delta(p^*) = 0$ и $\delta(q^*) = \frac{\delta(x^*)}{\left| \frac{\partial x}{\partial q} \right| \frac{|q^*|}{|x^*|}}$.

Вычислим $\frac{\partial x}{\partial q}$ и $\left| \frac{q^*}{x^*} \right|$.

$$\left| \frac{\partial x}{\partial q} \right| = \frac{1}{2 \sqrt{\left(\frac{p}{2}\right)^2 - q}} = 0.59805, \quad \min \left| \frac{q^*}{x^*} \right| = \frac{\ln 2}{1.8360} = 0.16396. \quad \text{Тогда } \max \delta(q^*) = \frac{0.0001}{0.59805 \cdot 0.16396} = 0.0010 = 0.1\%.$$

Таким образом, для того чтобы получить корни с четырьмя верными знаками, необходимо в исходном уравнении задать $\ln 2$ не менее чем с тремя верными знаками.

Пример. Для определения модуля Юнга по прогибу стержня прямоугольного сечения применяется формула $E = \frac{1}{4} \frac{l^3 P}{d^3 b S}$, где l - длина стержня, b и d - основание и высота попе-

речного сечения, S - стрела прогиба, P - нагрузка. С какой точностью следует измерить длину l и стрелу S , чтобы погрешность E не превышала 5.5% при условии, что P известна с точностью до 0.1%, величины b и d известны с точностью до 1%, $l \approx 50$ см, $S \approx 2.5$ см?

Применение формулы (1.4.2) для относительной погрешности E дает $\delta(E^*) = 3\delta(l^*) + \delta(P^*) + 3\delta(d^*) + \delta(b^*) + \delta(S^*)$. По условию задачи известно, что $\delta(E^*) = 5.5\% = 0.055$, $\delta(P^*) = 0.1\% = 0.001$, $\delta(b^*) = 1\% = 0.01$, $\delta(d^*) = 1\% = 0.01$. Тогда $3\delta(l^*) + \delta(S^*) = 1.4\%$. По принципу равных влияний на долю $3\delta(l^*)$ и $\delta(S^*)$ приходится по 0.7%. Тогда $\delta(S^*)$ равно 0.7%, то есть S надо измерять с точностью 0.7% = 0.007, l необходимо измерять точнее, $\delta(l^*) = 0.2\% = 0.002$. Отсюда $\Delta(S^*) = |S| \cdot \delta(S^*) = 2.5 \cdot 0.007 = 0.02$, а $\Delta(l^*) = |l| \cdot \delta(l^*) = 50 \cdot 0.002 = 0.1$, то есть $\begin{cases} S = 2.50 \div 0.02, \\ l = 50.0 \div 0.1. \end{cases}$

Задание № 1. Величина подъемной силы крыла самолета оценивается по формуле $F = c \rho v^2 S$, где S - площадь проекции крыла на горизонтальную плоскость, v - скорость натекания воздуха на крыло, ρ - плотность атмосферы на заданной высоте, α - угол атаки, отсчитываемый от направления нулевой подъемной силы, c - коэффициент, зависящий от формы крыла. Требуется вычислить F и $\delta(F^*)$ при заданных значениях c, α, ρ, v, S и заданных абсолютных или относительных значениях этих величин. (В таблице 1 для нечетных вариантов задано $\Delta(v^*)$ и $\delta(S^*)$, для четных $\delta(v^*)$ и $\Delta(S^*)$).

Таблица 1

Вариант	c	$\delta(c^*)$	α	$\delta(\alpha^*)$	ρ	$\delta(\rho^*)$	v	$\Delta(v^*)$ или $\delta(v^*)$	S	$\delta(S^*)$ или $\Delta(S^*)$
1	0.005	0.001	15	0.01	0.99	0.01	200	3	20	0.001
2	0.004	0.001	15	0.01	0.98	0.01	170	0.01	20	0.04
3	0.005	0.001	15	0.01	0.99	0.01	200	2	20	0.001
4	0.004	0.001	15	0.01	0.98	0.01	190	0.01	20	0.02
5	0.005	0.001	12	0.01	0.99	0.01	200	3	20	0.001
6	0.004	0.001	10	0.01	0.98	0.01	195	0.01	20	0.04
7	0.005	0.001	15	0.01	0.99	0.01	200	2	15	0.001
8	0.004	0.001	12	0.01	0.98	0.01	205	0.01	15	0.02
9	0.005	0.001	10	0.01	0.99	0.01	200	3	15	0.001
10	0.004	0.001	15	0.01	0.96	0.01	210	0.01	20	0.04
11	0.005	0.001	15	0.01	0.95	0.01	200	3	20	0.001
12	0.004	0.001	15	0.01	0.95	0.01	215	0.01	20	0.04
13	0.005	0.001	15	0.01	0.95	0.01	200	2	20	0.001
14	0.004	0.001	15	0.01	0.95	0.01	220	0.01	20	0.02
15	0.005	0.001	12	0.01	0.95	0.01	200	3	20	0.001
16	0.004	0.001	10	0.01	0.95	0.01	225	0.01	20	0.04
17	0.005	0.001	15	0.01	0.95	0.01	200	2	15	0.001
18	0.004	0.001	12	0.01	0.95	0.01	210	0.01	15	0.02
19	0.005	0.001	10	0.01	0.95	0.01	200	3	15	0.001
20	0.004	0.001	15	0.01	0.95	0.01	200	0.01	20	0.04
21	0.005	0.001	15	0.01	0.89	0.01	200	3	20	0.001
22	0.004	0.001	15	0.01	0.88	0.01	215	0.01	20	0.04
23	0.005	0.001	15	0.01	0.89	0.01	200	2	20	0.001
24	0.004	0.001	15	0.01	0.88	0.01	200	0.01	20	0.02
25	0.005	0.001	12	0.01	0.89	0.01	200	3	20	0.001
26	0.004	0.001	10	0.01	0.88	0.01	220	0.01	20	0.04
27	0.005	0.001	15	0.01	0.89	0.01	200	2	15	0.001
28	0.004	0.001	12	0.01	0.88	0.01	220	0.01	15	0.02
29	0.005	0.001	10	0.01	0.89	0.01	200	3	15	0.001
30	0.004	0.001	15	0.01	0.86	0.01	200	0.01	20	0.04

Образец выполнения задания № 1

Пусть $c = 0.006$, $\delta(c^*) = 0.001$, $\alpha = 13$, $\delta(\alpha^*) = 0.01$, $\rho = 0.88$, $\delta(\rho^*) = 0.01$, $v = 225$, $\Delta(v^*) = 3$, $S = 17$, $\delta(S^*) = 0.001$. Относительная погрешность функции $F = c\alpha\rho v^2 S$ равна $\delta(F^*) = \delta(c^*) + \delta(\alpha^*) + \delta(\rho^*) + 2\delta(v^*) + \delta(S^*)$, абсолютная $\Delta(F^*) = |F| \cdot \delta(F^*)$, $\delta(F^*) = 0.001 + 0.001 + 0.01 + 2 \cdot \frac{3}{225} + 0.001 = 0.040 = 4.0\%$.

Так как относительная погрешность велика, то значение функции следует вычислять не более чем с двумя – тремя знаками (с запасом), то есть $F = 0.006 \cdot 13 \cdot 0.88 \cdot 225^2 \cdot 17 = 590 \cdot 10^2$ (а не 59073 и не 59000, так как тогда было бы $\Delta(F^*) = 0.5$ и $\delta(F^*) = 0.0008\%$). Отсюда $\Delta(F^*) = 590 \cdot 10^2 \cdot 0.040 = 24 \cdot 10^2$. Поскольку $\Delta(F^*) \approx 10^3$, то результат целесообразно округлить до двух знаков. Окончательно, $F = 5.9 \cdot 10^4$, $\Delta(F^*) = 0.2 \cdot 10^4$, $\delta(F^*) = 4.0\%$.

Задание № 2. Найти абсолютную погрешность вычисления функции при заданных значениях аргументов.

$$1. S = 1/2 ab \sin C$$

	1	2	3
a	$17.25 \div 0.01$	$2.1 \div 0.05$	$6.089 \div 0.003$
b	$34.725 \div 0.005$	$18.4 \div 0.03$	$0.121 \div 0.001$
S	$18.3 \div 0.01$	$2.06 \div 0.005$	$167.18 \div 0.01$

$$2. V = 1/3 h(S_1 + S_2 + \sqrt{S_1 S_2})$$

	1	2	3
h	$21.7 \div 0.02$	$3.45 \div 0.01$	$0.956 \div 0.004$
S_1	$9.272 \div 0.001$	$17.654 \div 0.0005$	$5.322 \div 0.0005$
S_2	$11.01 \div 0.005$	$2.19 \div 0.02$	$1.315 \div 0.0004$

$$3. V = 1/6 \pi h^3 + 1/2 \pi(r_1^2 + r_2^2)h$$

	1	2	3
h	$1.5 \div 0.005$	$0.21 \div 0.03$	$3.78 \div 0.002$
r_1	$6.28 \div 0.02$	$11.615 \div 0.001$	$139.96 \div 0.006$
r_2	$5.137 \div 0.0003$	$11.09 \div 0.005$	$125.3 \div 0.01$

$$4. A = \arccos \frac{b^2 + c^2 - a^2}{2bc}$$

	1	2	3
a	$3.456 \div 0.001$	$2.5 \div 0.03$	$86.2 \div 0.05$
b	$7.12 \div 0.004$	$0.854 \div 0.0005$	$44.831 \div 0.001$
c	$9.7 \div 0.1$	$1.16 \div 0.004$	$67.18 \div 0.006$

$$5. y = Ae^{-ax} \sin(\omega x + \phi_0)$$

	1	2	3
A	$49.83 \div 0.01$	$0.85 \div 0.003$	$4.523 \div 0.003$
a	$2.35 \div 0.001$	$0.536 \div 0.002$	$0.014 \div 0.0005$
ω	$11.7 \div 0.1$	$3.7 \div 0.02$	$1.76 \div 0.002$
ϕ_0	$3.147 \div 0.0001$	$0.7071 \div 0.00005$	$54 \div 0.5$
x	$1.78 \div 0.001$	$0.0465 \div 0.0001$	$3.14 \div 0.005$

$$6. \ y^2 = x^2 \left(\frac{a+x}{a-x} \right)$$

	1	2	3
a	$2.435 \div 0.005$	$72 \div 0.3$	$1.84 \div 0.006$
x	$1.27 \div 0.01$	$3.374 \div 0.002$	$0.8345 \div 0.0004$

$$7. \ y = a \cdot \operatorname{tg} \varphi + l \sin \varphi$$

	1	2	3
a	$3.889 \div 0.001$	$145.5 \div 0.08$	$27.3 \div 0.04$
l	$0.8454 \div 0.0002$	$28.6 \div 0.1$	$0.93 \div 0.001$
φ	$0.643 \div 0.0005$	$0.1736 \div 0.00005$	$1.73 \div 0.03$

$$8. \ r = \frac{a}{\varphi} \left(\frac{\sqrt{1 + \varphi^2}}{\varphi} \right)^3$$

	1	2	3
a	$1.245 \div 0.001$	$15.8 \div 0.03$	$3.45 \div 0.001$
φ	$0.3746 \div 0.0001$	$0.9848 \div 0.0003$	$1.8387 \div 0.00005$

$$9. \ L = a \sqrt{1 + \lambda^2 - 2\lambda \cos t}$$

	1	2	3
a	$9.14 \div 0.005$	$3.6 \div 0.04$	$23.76 \div 0.003$
λ	$1.2 \div 0.05$	$0.2518 \div 0.0001$	$0.85 \div 0.006$
t	$0.56 \div 0.01$	$0.34 \div 0.005$	$1.8 \div 0.01$

$$10. \ y = a \ln \frac{a + \sqrt{a^2 - x^2}}{x}$$

	1	2	3
a	$5.93 \div 0.005$	$2.786 \div 0.0006$	$0.75 \div 0.003$
x	$3.1415 \div 0.0001$	$0.734 \div 0.004$	$0.0761 \div 0.0001$

$$11. \ S = \frac{\pi a^2}{2} + \pi l^2$$

	1	2	3
a	$52.34 \div 0.01$	$0.0545 \div 0.00005$	$0.65 \div 0.02$
l	$2.0435 \div 0.0001$	$0.82 \div 0.01$	$6.3 \div 0.02$

$$12. \ V = \frac{2\pi R^2 h}{3}$$

	1	2	3
h	$0.285 \div 0.0002$	$0.2731 \div 0.0002$	$5.843 \div 0.001$
R	$0.64 \div 0.004$	$10.8 \div 0.02$	$4.17 \div 0.001$

$$13. \ H = h + \frac{hr}{R-r}$$

	1	2	3
h	$8.23 \div 0.005$	$0.5324 \div 0.0001$	$2.5 \div 0.03$
r	$0.281 \div 0.001$	$0.032 \div 0.003$	$1.7 \div 0.01$
R	$11.7 \div 0.05$	$1.81 \div 0.005$	$31.456 \div 0.002$

$$14. R^2 = a^2 + \left(\frac{a^2 - b^2 - h^2}{2h} \right)^2$$

	1	2	3
a	$23.76 \div 0.003$	$3.236 \div 0.002$	$17.3 \div 0.03$
b	$5.8 \div 0.02$	$1.315 \div 0.0004$	$4.431 \div 0.001$
h	$3.45 \div 0.001$	$1.57 \div 0.006$	$0.28 \div 0.006$

$$15. y = \frac{x_1 + \lambda x_2}{1 + \lambda}$$

	1	2	3
x_1	$7.751 \div 0.0005$	$29.72 \div 0.01$	$3.14 \div 0.005$
x_2	$0.12 \div 0.01$	$78 \div 0.3$	$0.6384 \div 0.0002$
λ	$1.0 \div 0.05$	$0.512 \div 0.0005$	$2.1951 \div 0.0001$

$$16. S = \frac{ae - bd}{\sqrt{a^2 + b^2}}$$

	1	2	3
a	$2.4 \div 0.02$	$0.121 \div 0.0002$	$10.536 \div 0.001$
b	$3.35 \div 0.005$	$5.8 \div 0.02$	$7.18 \div 0.006$
e	$8.235 \div 0.001$	$21.1 \div 0.1$	$0.8 \div 0.1$
d	$0.258 \div 0.0002$	$0.1756 \div 0.0001$	$0.4834 \div 0.0001$

$$17. K = \frac{1}{a} \frac{\varphi^2 + 2}{(\varphi^2 + 1)^{\frac{3}{2}}}$$

	1	2	3
a	$0.3575 \div 0.0002$	$5.12 \div 0.01$	$3.6 \div 0.04$
φ	$28.6 \div 0.1$	$1.27 \div 0.002$	$7.221 \div 0.003$

$$18. F = \frac{\ln v + \sqrt{\arcsin u}}{u^2 + 5e^v}$$

	1	2	3
u	$0.184 \div 0.006$	$0.357 \div 0.0005$	$0.63 \div 0.02$
v	$0.3575 \div 0.0002$	$0.25 \div 0.003$	$0.0562 \div 0.0001$

$$19. y = 3k \cdot \operatorname{arctg} \frac{1}{2^x + 3}$$

	1	2	3
k	$81.26 \div 0.01$	$341.17 \div 0.03$	$4.2 \div 0.05$
x	$0.142 \div 0.0003$	$0.642 \div 0.0005$	$0.28 \div 0.006$

$$20. y = k \left(\cos \frac{2}{\sqrt[3]{x}} - 1 \right)^3$$

	1	2	3
k	$4.218 \div 0.001$	$10.5 \div 0.1$	$1.0397 \div 0.00005$
x	$0.15 \div 0.005$	$0.0545 \div 0.0001$	$81.26 \div 0.02$

$$21. y = (a - b)^2 \operatorname{ch}(x)$$

	1	2	3

<i>a</i>	3.0567÷0.0001	0.678÷0.003	5.72÷0.02
<i>b</i>	28.3÷0.02	7.45÷0.01	3.2817÷0.0001
<i>x</i>	6.3÷0.02	0.82÷0.01	4.2÷0.05

22. $r = \sqrt{\frac{(p-a)(p-b)(p-c)}{p}}, p = \frac{a+b+c}{2}$

	1	2	3
<i>a</i>	4.16÷0.005	7.27÷0.01	12.163÷0.002
<i>b</i>	5.205÷0.001	6.3÷0.04	3.244÷0.001
<i>c</i>	2.39÷0.01	4.198÷0.0005	9.3÷0.1

23. $r = \frac{2a^2\rho^3}{c^4 - a^4 + 3\rho^4}$

	1	2	3
<i>a</i>	4.22÷0.004	3.4÷0.1	5.82÷0.005
<i>c</i>	7.5÷0.05	8.13÷0.002	8.632÷0.003
ρ	5.205÷0.001	1.315÷0.0005	0.14÷0.01

24. $y = \frac{\sqrt{1-2a}-3}{\sqrt{b}-2}$

	1	2	3
<i>a</i>	0.28÷0.006	0.156÷0.0005	0.08÷0.01
<i>b</i>	10.536÷0.002	7.18÷0.01	31.110÷0.0005

25. $v = \sin(a^3) \sin \frac{b+1}{b}$

	1	2	3
<i>a</i>	0.235÷0.0003	0.318÷0.0002	1.8÷0.01
<i>b</i>	48.39÷0.001	0.46÷0.002	3.108÷0.0003

26. $R = \frac{\ln \sin(3a)}{(6b - \pi)^2}$

	1	2	3
<i>a</i>	0.44÷0.03	0.772÷0.0005	1.49÷0.01
<i>b</i>	9.542÷0.001	4.218÷0.001	29.725÷0.0005

27. $v = \frac{\cos(3x_1) - \cos x_2}{\operatorname{tg}^2(2x_2)}$

	1	2	3
x_1	3.35÷0.01	6.17÷0.02	5.441÷0.0005
x_2	0.383÷0.0002	0.891÷0.0005	1.3÷0.01

28. $S = \frac{\pi}{4}(D^2 - d^2)$

	1	2	3
<i>d</i>	17.29÷0.005	28.3÷0.1	32.14÷0.02
<i>D</i>	72÷0.3	42.08÷0.005	116.929÷0.0005

$$29. V = \pi R^2 \frac{h_1 + h_2}{2}$$

	1	2	3
h_1	1.29÷0.005	11.3461÷0.0001	1.84÷0.006
h_2	2.316÷0.001	6.83÷0.03	0.8345÷0.0004
R	3.274÷0.0005	9.1÷0.1	32.7÷0.04

$$30. K = \frac{a \sin b}{c - a \cos b}$$

	1	2	3
a	2.45÷0.01	7.834÷0.001	0.15÷0.01
b	0.6294÷0.0001	0.11113÷0.00005	1.29÷0.005
c	17.3÷0.03	11.99÷0.005	2.4÷0.02

Образец выполнения задания № 2

$$\text{Дано } y^2 = \frac{x^3}{a-x}$$

	1
a	1.1753÷0.0003
x	0.318÷0.0005

Абсолютные погрешности аргументов $\Delta(a^*) = 0.0003$, $\Delta(x^*) = 0.0005$. Относительные погрешности будут равны: $\delta(a^*) = \frac{\Delta(a^*)}{|a^*|} = 0.0003 = 0.03\%$, $\delta(x^*) = \frac{\Delta(x^*)}{|x^*|} = 0.0016 = 0.16\%$.

По формуле (1.4.2) получим $\delta(y^*) = \sum_{j=1}^2 v_j \delta(x_j) = v_1 \delta(x_1) + v_2 \delta(x_2)$, где

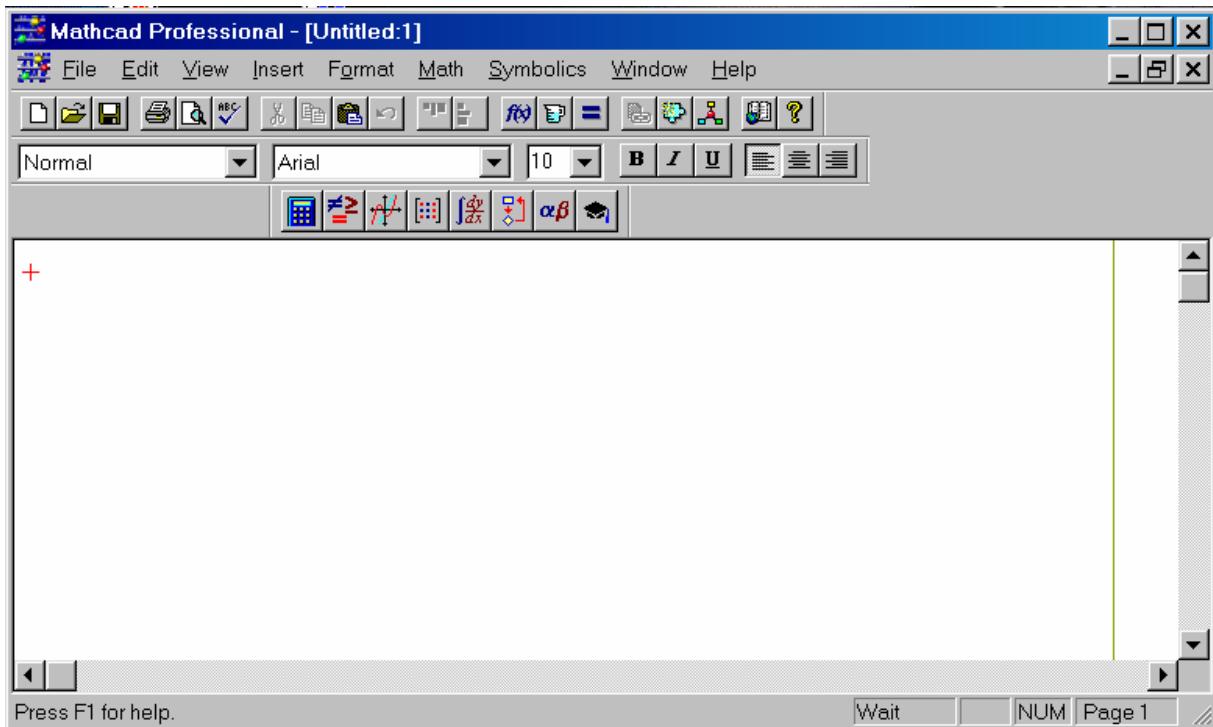
$$v_1 = \frac{|a^*| \left| \frac{\partial y}{\partial a} \right|}{|y^*|} = \frac{|3a^* - 2x^*|}{|2(a^* - x^*)|}, \quad v_2 = \left| \frac{a^*}{2(a^* - x^*)} \right|$$

$\delta(y^*) = 0.6855 \cdot 0.0003 + 1.6855 \cdot 0.0016 = 0.0029 = 0.29\%$. Само значение функции равно $y^* = 0.1937$. $\Delta(y^*) = |y^*| \cdot \delta(y^*) = 0.0006$. Следует округлить значение функции до трех значащих цифр. Итак, $y^* = 0.194$, $\delta(y^*) = 0.29\%$, $\Delta(y^*) = 0.0006$.

Познакомимся теперь с одним из самых популярных и весьма распространенных математических пакетов – пакетом Mathcad. В нем возможна запись формул в привычном виде на экране, редактирование текстовой и графической информации, визуализация результатов вычислений в виде двух- и трехмерных графиков. Пакет Mathcad – это полноценное Windows–приложение, в него интегрирован весьма мощный математический аппарат, позволяющий решать вычислительные задачи без вызова внешних процедур.

По ходу выполнения лабораторных работ по курсу “Численные методы” научимся быстро и легко решать в среде Mathcad простейшие математические задачи. Подробная характеристика и описание пакета даны в специальной литературе справочного характера. Рассмотрим конфигурацию Mathcad 7 Professional, входящего в математическое обеспечение вычислительной лаборатории кафедры вычислительной и прикладной математики БГТУ.

Пакет запускается стандартным образом – двойным щелчком левой кнопки мыши по пиктограмме Mathcad. Происходит автоматическая загрузка всех программ, необходимых для работы пакета, затем открывается окно приложения, имеющее стандартную Windows–структуру.



В первой строке приведено имя приложения – Mathcad Professional, затем имя рабочего файла. Справа в строке находятся три функциональные кнопки: свернуть, развернуть на полный экран, закрыть окно приложения. Вторая строка – это строка меню, состоящая как из стандартных для Windows-приложений пунктов: Файл (File), Редактировать (Edit), Просмотр (View), Формат (Format), Окно (Window), Помощь (Help), так и специфических, созданных специально для пакета Mathcad: Вставка (Insert), Математика (Math), Символьные операции (Symbolics). Щелчок по любому пункту меню открывает подменю со списком доступных операций.

Следующая третья строка – панель инструментов. Для того чтобы узнать, что делает кнопка панели, достаточно подвести к ней курсор, и появится строка сообщения. Прямо под панелью инструментов располагается панель шрифтов. Она содержит шаблоны выбора и кнопки, используемые для задания характеристик шрифтов в уравнениях и тексте.

Наконец, в последней строке расположена панель инструментов для выполнения математических операций. Каждая кнопка из этой панели открывает палитру символов:



Общие арифметические операторы;



Знаки отношения;



Различные двух -и трехмерные графики;



Матричные и векторные операции;



Производные, интегралы и пределы. Ряды и произведения;



Программные структуры;



Греческие буквы;



Ключевые слова символьных вычислений.

Любая из описанных панелей может быть помещена в произвольное место окна. При этом панель инструментов и панель математических операций будут автоматически перестраиваться соответственно месту экрана, на которое они попадают.

Под панелями инструментов находится окно рабочего документа Mathcad. Содержимое этого окна можно редактировать, печатать, сохранять в файлах на диске и так далее. Для просмотра всего пространства служат две полосы прокрутки: вертикальная и горизонтальная.

Содержание первой лабораторной работы таково, что Mathcad можно использовать как простой калькулятор. Опишем пример подобных вычислений, определив $\delta(F^*)$ из образца выполнения задания № 1.

После щелчка в любом месте рабочего документа появляется небольшой крестик. С этого места в рабочем документе будет помещаться весь ввод с клавиатуры. Введем символы в следующей последовательности: $0.001 + 0.01 + 0.01 + 2 * 3 / 225 + 0.001 =$. После набора знака $=$ Mathcad вычисляет выражение и выводит результат. На экране отобразится формула в точности в том виде, как ее пишут на доске: $0.001 + 0.01 + 0.01 + 2 \cdot \frac{3}{225} + 0.001 = 0.040$.

Для удаления любой цифры в выражении надо щелкнуть мышью справа внизу от нее и нажать клавишу $\langle \uparrow \rangle$ - (Backspace). Для удаления всего выражения щелкаем мышью по любому месту в выражении и нажимаем клавишу $\langle \text{Пробел} \rangle$ -(Space) до тех пор, пока все оно не будет выделено угловой синей рамкой. После этого нажимаем клавиши (Backspace) и $\langle \text{Удалить} \rangle$ -(Delete). Выражение исчезнет с экрана.

Чтобы определить любую переменную, нужно напечатать ее имя, затем набрать символ $\langle :\rangle$. В рабочем документе этот символ отобразится, как знак присваивания $\langle := \rangle$. После этого нужно набрать значение, присваиваемое набранной переменной, например, $v := 50.5$, и либо щелкнуть мышкой по свободному месту вне поля ввода, либо нажать клавишу $\langle \downarrow \rangle$ - (Enter).

Аналогично вычисляется значение функции. Для этого нужно начать с определения функции, например, набрав $S[0 : 2 \langle \rightarrow \rangle S(v, t) : S[0 \langle \text{Space} \rangle + v * t \langle \rightarrow \rangle t : 4.25 \langle \rightarrow \rangle]$. На экране отобразятся следующие выражения: $S_0 := 2$ $S(v, t) := S_0 + v \cdot t$ $t := 4.25$. Нижние индексы вводятся нажатием клавиши $\langle [\rangle$. Что бы ни печаталось после $[$, все остается в нижнем индексе, пока не будет нажата клавиша $\langle \text{Space} \rangle$, чтобы выйти оттуда.

Теперь можно использовать определенную функцию $S(v, t)$, чтобы вычислить ее значение для произвольных значений аргументов. Чтобы вычислить функцию для $v = 50.5$, $t = 4.25$, достаточно напечатать $S(v, t) =$. Mathcad возвратит соответствующее значение $S(v, t) = 216.625$. Этого же результата можно добиться, подставляя числовые значения аргументов в функцию: $S(50.5, 4.25) = 216.625$.

Определим теперь дискретные значения аргументов v и t . Это делается следующим образом. Сначала печатается имя дискретного аргумента. Это должно быть простое имя. Никакие нижние индексы или функциональные определения недопустимы. Затем набирается двоеточие и диапазон значений, например, $v : 1; 3$. На экране отобразится: $v := 1 \dots 3$. Это самое простое определение дискретного аргумента. Здесь величина шага Δv равна единице. Mathcad допускает дискретные аргументы со значениями, расположенными от любого числа до любого другого значения с произвольным постоянным шагом. Например, введем $t : 0, 3; 6$.

Здесь 0 - первое значение, принимаемое аргументом t , 3 - второе значение аргумента t в диапазоне, 6 - последнее значение. Величина шага определяется автоматически:

$\Delta t = 3 - 0 = 3$. На экране появится выражение $t := 0,3\dots6$. Далее вводим $M[v, t / 3 : S(v, t) \rightarrow M =$. На экране отобразится:

$$v := 1\dots3 \quad t := 0,3\dots6 \quad M_{v, \frac{t}{3}} := S(v, t)$$

$$M = \begin{bmatrix} 0 & 0 & 0 \\ 2 & 5 & 8 \\ 2 & 8 & 14 \\ 2 & 11 & 20 \end{bmatrix}$$

Mathcad содержит восемь переменных, значения которых определяются сразу после запуска программы. Это предопределенные или встроенные переменные. Одна из таких переменных - ORIGIN - определяет индекс первого элемента массива. По умолчанию в Mathcad координаты векторов и строки матриц нумеруются с нуля (ORIGIN := 0). Эти восемь переменных можно переопределять в любом месте рабочего документа. Как и в электронных таблицах, любое изменение содержимого рабочего документа Mathcad вызывает обновление всех зависимых результатов и перерисовку графиков.

Построим теперь несколько графиков функций одной и нескольких переменных, что потребуется в следующих лабораторных работах. В последней строке панели инструментов есть кнопка различных двух- и трехмерных графиков. Щелчок по ней открывает дополнительное меню:



Наиболее часто употребляемые типы графиков:



X-Y Plot – графики функций одной переменной в декартовых координатах;



Polar Plot - графики функций одной переменной в полярных координатах;



Surface Plot – графики поверхностей (функций двух переменных) в декартовых координатах;



Contour Plot – графики линий уровня функций двух переменных в декартовых координатах;



3D Scatter Plot – изображения точек в трехмерном пространстве в декартовых координатах;

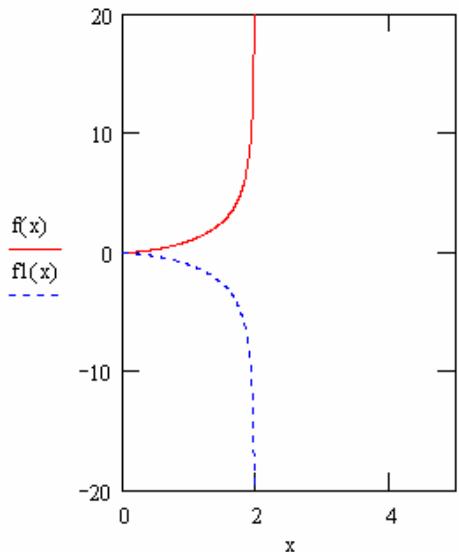


3D Bar Chart – трехмерные гистограммы.

Все перечисленные графики, как это принято во всех Windows-приложениях, можно вызвать из главного меню выбором соответствующих операций: Вставка (Insert) – Графика (Graph).

Построим график функции из образца выполнения задания № 2. Это функция $y^2 = \frac{x^3}{a-x}$, ее график называется циссоидой. Полярное уравнение этой кривой имеет вид $r = a \cdot \operatorname{tg}\varphi \sin\varphi$. Для построения графика в декартовой и полярной системе необходима явная зависимость $y = f(x)$ или $r = r(\varphi)$. Определим $y = \pm \sqrt{\frac{x^3}{a-x}}$. Для двух ветвей графика

необходимо определить две разные функции. Введем с клавиатуры $a := 2$; $f(x) := \sqrt{x^3/a} - x$; $f1(x) := -f(x)$. Теперь выберем пункт **Декартов график** из меню **Графика**. Появляется пустой график с полями ввода для выражений, отображаемых по осям графика. Множество точек, из которых состоит график, определяется дискретными аргументами. В поле ввода под осью абсцисс нужно ввести имя аргумента x , а в поле ввода напротив середины оси ординат ввести $f(x)$ и через запятую $f1(x)$. Оставшиеся поля предназначены для ввода границ на полях (максимального и минимального значений x и $f(x)$). Если оставить эти поля пустыми, Mathcad автоматически заполнит их при создании графика. После щелчка вне графика Mathcad вычисляет и строит точки обоих графиков. На экране появится



$$a := 2 \quad f(x) := \sqrt{\frac{x^3}{a}} \quad f1(x) := -f(x).$$

Двойной щелчок в любом месте графика вызывает панель форматирования. Здесь можно установить все доступные характеристики графика. Те же действия выполняются при построении графика в полярной системе координат. С учетом периодичности и нечетности синуса и тангенса для отображения обеих ветвей графика надо набрать $r(\phi) := -a * \text{atan}(\phi - \pi) * \text{asin}(\phi - \pi)$, затем щелкнуть по пиктограмме полярного графика; по оси аргумента в пустом графике ввести ϕ , по оси функции $r(\phi)$. На экране отразится $r(\phi) := -a \cdot \text{atan}(\phi - \pi) \cdot \text{asin}(\phi - \pi)$. Число π входит в восемь предопределенных переменных. В численных расчетах Mathcad использует значение π с учетом 15 значащих цифр. Чтобы ввести π , надо нажать [Ctrl]P.

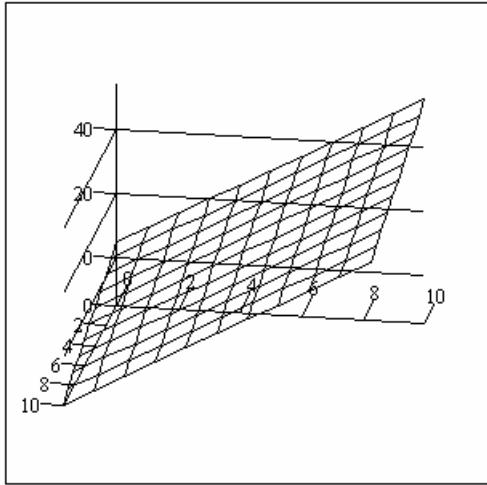
Чтобы создать график функции двух переменных, необходимо сначала образовать матрицу, содержащую значения этой функции, а затем построить поверхностный график этой матрицы. Построим, например, плоскость $z = 5 - 2x + 5y$ и часть гиперболического параболоида $z = xy$. Для

этого необходимо набрать

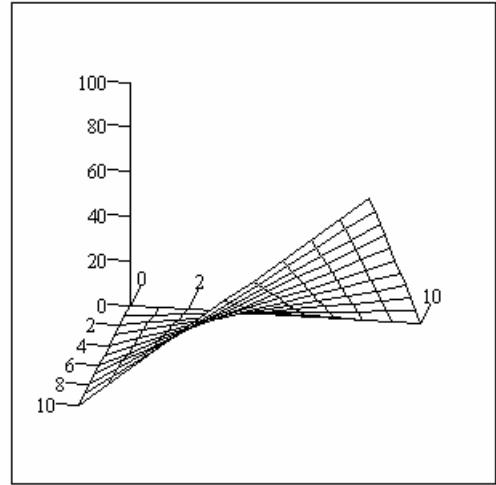
$$i : 0..10 \rightarrow j : 0..10 \rightarrow x[i : i \rightarrow] y[j : j \rightarrow] z(x, y) := 5 - 2 * x + 5 * y \rightarrow M[i, j : z(x[i \rightarrow], y[j \rightarrow]) \rightarrow] \\ y[j \rightarrow] \rightarrow z1(x, y) := x * y \rightarrow M1[i, j : z1(x[i \rightarrow], y[j \rightarrow])]$$

Далее, нажав на кнопку графика поверхности в панели графиков , введем в первом графике в поле ввода матрицу M , во втором - матрицу $M1$. На экране монитора получим следующую картину:

$$i := 0..10 \quad j := 0..10 \quad x_i := i \quad y_j := j \quad z(x, y) := 5 - 2 \cdot x + 5 \cdot y \\ M_{i,j} := z(x_i, y_j) \quad z1(x, y) := x \cdot y \quad M1_{i,j} := z1(x_i, y_j)$$



M



M1

Заметим, что при данном выборе дискретных аргументов x_i и y_j поверхности строятся только в первом октанте. Выбором соответствующей формулы типа $x_i = f(i)$ всегда можно добиться равномерного расположения точек x_i в нужном месте координатной оси. Построим в заключение поверхность функции плотности вероятности двумерного нормального закона и графики линий уровня этой функции. Функция плотности кроме аргументов x, y зависит от пяти параметров: $m_x, m_y, \sigma_x, \sigma_y, r_{xy}$ и имеет вид

$$f(x, y) = \frac{1}{2\pi\sigma_x\sigma_y\sqrt{1-r_{xy}^2}} \exp\left\{-\frac{1}{2(1-r_{xy}^2)}\left[\frac{(x-m_x)^2}{\sigma_x^2} - \frac{2r_{xy}(x-m_x)(y-m_y)}{\sigma_x\sigma_y} + \frac{(y-m_y)^2}{\sigma_y^2}\right]\right\},$$

а ее линии уровня определяются уравнением $\frac{(x-m_x)^2}{\sigma_x^2} - \frac{2r_{xy}(x-m_x)(y-m_y)}{\sigma_x\sigma_y} + \frac{(y-m_y)^2}{\sigma_y^2} = k$, $k = \text{const}$ и получаются сечением поверхности

$f(x, y)$ плоскостями, параллельными координатной плоскости XOY . Итак, введем

$\sigma x : 5 \rightarrow \sigma y : 5 \rightarrow rxy : 0.3 \rightarrow mx : 15 \rightarrow my : 20$

$$f1(x, y) : (x - mx)^2 / \sigma x^2 \langle \text{Space} \rangle - 2 * rxy * (x - mx) * (y - my) / \sigma x * \sigma y \langle \text{Space} \rangle$$

$$+ (y - my)^2 / \sigma y^2 \langle \downarrow i : 0; 30 \rightarrow j : 0; 30 \rightarrow x[i : i \rightarrow] y[j : j \downarrow] \rangle$$

$$M2[i, j : 1 / 2 * [\text{Ctrl}]P * \sigma x * \sigma y * \sqrt{1 - rxy^2} * \exp(-f1(x[i \rightarrow], y[j \rightarrow])) /$$

$$2 * (1 - rxy^2)) \rightarrow M3[i, j : f1(x[i \rightarrow], y[j \rightarrow])]$$

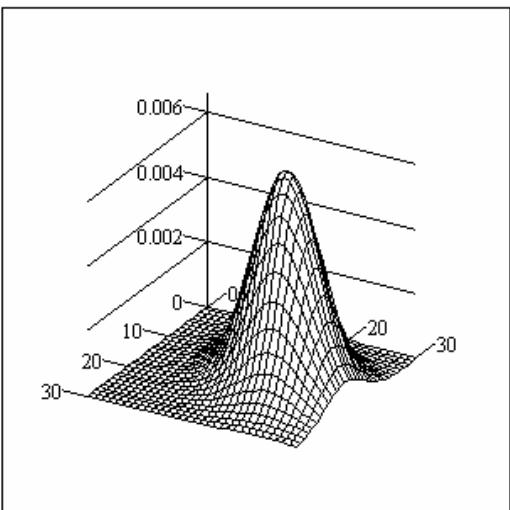
На экране отобразится:

$$\sigma x := 5 \quad \sigma y := 5 \quad rxy := 0.3 \quad mx := 15 \quad my := 20$$

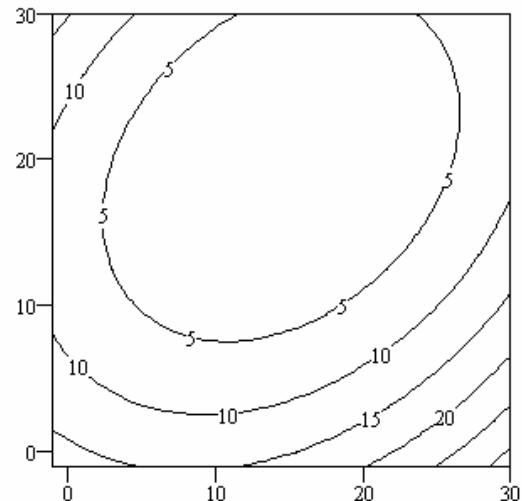
$$f1(x, y) := \frac{(x - mx)^2}{\sigma x^2} - 2 * rxy * (x - mx) * \frac{(y - my)}{\sigma x * \sigma y} + \frac{(y - my)^2}{\sigma y^2}$$

$$i := 0 \dots 30 \quad j := 0 \dots 30 \quad x_i := i \quad y_j := j$$

$$M2_{i,j} := \frac{1}{2 * \pi * \sigma x * \sigma y * \sqrt{1 - rxy^2}} \exp\left(\frac{-f1(x_i, y_j)}{2 * (1 - rxy^2)}\right) \quad M3_{i,j} := f1(x_i, y_j)$$



M2



M3

С помощью панели форматирования первый график повернут на 30° , и на нем убраны невидимые линии, оси второго графика оцифрованы по x и y в пределах от 0 до 30.

1.7. Корректность вычислительной задачи

Анализ важнейших требований, предъявляемых к различным прикладным задачам, приводит к понятию корректности математической задачи.

Вычислительная задача называется корректной, если выполняются следующие три требования: а) решение этой задачи $y \in Y$ существует при любых $x \in X$, б) это решение единственное, в) решение устойчиво по отношению к малым возмущениям входных данных. Если не выполнено хотя бы одно из условий, задача называется некорректной.

Существование решения - естественное требование. Отсутствие решения свидетельствует либо о непригодности принятой математической модели, либо о неправильной постановке самой математической модели.

Неединственность - неприятное свойство вычислительной задачи. Ее причиной, помимо уже перечисленных условий, может быть естественное свойство решаемой задачи. Неединственность может быть ликвидирована введением некоторых дополнительных ограничений на решение. Иногда за решение задачи принимается множество решений.

Решение y называется устойчивым по входным данным x , если y зависит от x непрерывным образом. Строго формальное определение устойчивости решения похоже на определения предела функции: если для любого $\varepsilon > 0$ найдется такое $\delta = \delta(\varepsilon) > 0$, что для любого $x_i \in X$, удовлетворяющего условию $\Delta(x_i) < \delta$, найдется соответствующее $y_i \in Y$, такое, что $\Delta(y_i) < \varepsilon$, решение будет устойчиво. Иными словами, для устойчивости вычислительной задачи ее решение теоретически можно найти со сколь угодно высокой точностью ε , если обеспечена высокая точность δ исходных данных. Неустойчивость решения y означает, что $\exists \varepsilon_0 > 0$, что какое бы малое $\delta > 0$ ни было задано, найдутся такие данные x_i , что $\Delta(x_i) < \delta$, но $\Delta(y_i) > \varepsilon_0$.

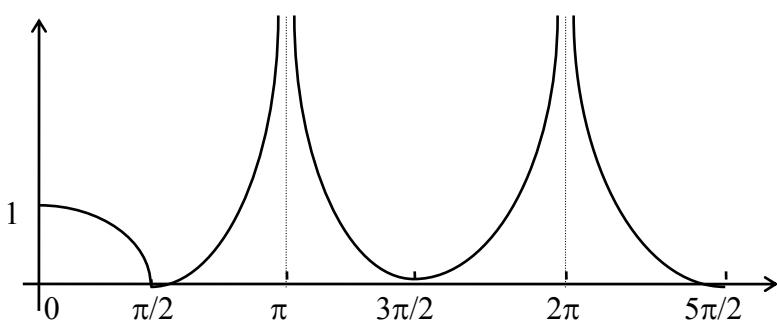
1.8. Обусловленность вычислительной задачи

Под обусловленностью вычислительной задачи понимают ее чувствительность к малым погрешностям входных данных. **Задачу называют хорошо обусловленной, если малым погрешностям входных данных отвечают малые погрешности решения, и плохо обусловленной, если возможны сильные изменения решения.** Для измерения количественной стороны обусловленности используют число обусловленности. Грубо говоря, это коэффициент возможного возрастания погрешностей в решении по отношению к вызвавшим их погрешностям входных данных, то есть $\Delta(y_i) = v_\Delta \cdot \Delta(x_i)$, где v_Δ - абсолютное число обусловленности. Если $\delta(y_i) = v_\delta \cdot \delta(x_i)$, то v_δ - относительное число обусловленности. Для плохо обусловленной задачи $v >> 1$.

Пример. Вычисление значений функции одного переменного. При этом

$$v_\Delta = |f'(x)| \text{ и } v_\delta = \frac{|x| \cdot |f'(x)|}{|f(x)|} \text{ в силу формул} \begin{cases} \Delta(y^*) = |f'(x)| \Delta(x^*) \text{ и} \\ \delta(y^*) = \frac{|x| \cdot |f'(x)|}{|f(x)|} \delta(x^*) \end{cases} \text{Вычислим } y = \sin x.$$

В этом случае $v_\Delta = |\cos x| \leq 1$, что говорит о хорошей абсолютной обусловленности этой задачи при всех x . Однако если важен результат с определенным числом верных знаков, то нужно использовать относительную обусловленность.



Тогда $v_\delta = |x \cdot \operatorname{ctg} x|$. График функции $y = |x \cdot \operatorname{ctg} x|$ приведен слева. Так как $v_\delta \rightarrow \infty$ при $x \rightarrow \pi k$, то при $x \approx \pi k$ задача обладает плохой относительной обусловленностью, хотя мала абсолютная по-

грешность значения $y^* = \sin x^*$. Если же значение $|x|$ очень велико, то $v_\delta >> 1$. Например, для ЭВМ РС при $\varepsilon_m \approx 10^{-7}$ и при $|x| \approx 10^7$ одна только абсолютная ошибка $\Delta(x^*) \approx 1$.

1.9. Вычислительные методы, их классификация

Методы, которые используются в вычислительной математике для преобразования задач к виду, удобному для реализации на ЭВМ, и позволяют конструировать вычислительные алгоритмы, называются вычислительными.

Они разделяются на следующие классы:

- 1) методы эквивалентных преобразований;
- 2) методы аппроксимации;
- 3) прямые (точные) методы;
- 4) итерационные методы;
- 5) методы статистических испытаний (Монте-Карло).

Поясним классификацию подробнее.

Методы эквивалентных преобразований позволяют заменить исходную задачу другой, имеющей то же решение. Это целесообразно, если новая задача проще исходной или обладает лучшими свойствами, или для нее, например, существует уже готовый метод и программа, а для исходной задачи их надо создавать заново.

Пример. Задача отыскания корней уравнения $y = f(x) = 0$ может быть сведена к эквивалентной задаче поиска точки глобального минимума функции $\Phi(x) = (f(x))^2$.

Методы аппроксимации приближают исходную задачу другой, решение которой в оговоренном смысле близко к решению исходной задачи. Погрешности, возникающие при такой замене, называются погрешностями аппроксимации. Как правило, аппроксимирующая задача содержит параметры, позволяющие регулировать величину погрешности аппроксимации.

Пример. Формула прямоугольников вычисления определенного интеграла:

$$I = \int_a^b f(x) dx \Rightarrow I_{\text{выч}} = h \sum_{i=1}^n f\left(a + \left(i - \frac{1}{2}\right)h\right), \text{ где } h = \frac{b-a}{n} \text{ - параметр, регулирующий точность.}$$

Прямые методы позволяют получить решение исходной задачи после выполнения конечного числа элементарных операций.

Пример. $x^2 + bx + c = 0$, метод вычисления корней $x_{1,2} = \left(-b \pm \sqrt{b^2 - 4ac}\right)/2$. Здесь элементарными считаются четыре арифметические операции и операция извлечения квадратного корня. Элементарной операцией метода может быть объявлена любая другая операция, необходимо, конечно, чтобы ее выполнение было существенно проще решения всей задачи.

Итерационные методы - специальные методы, приспособленные для построения последовательных приближений к точному решению задачи. Для получения каждого из последующих приближений выполняют однотипный набор действий с использованием ранее найденных приближений - итераций («iteratio»- повторение). Неограниченное продолжение этого процесса позволяет построить бесконечную последовательность приближений к решению - *итерационную последовательность*. Если она сходится к решению, то говорят, что итерационный метод сходится.

Множество начальных приближений, для которых метод сходится, называется областью сходимости метода. Практическая реализация итерационных методов всегда связана с необходимостью выбора критерия окончания итерационного процесса.

Пример. Вычислим \sqrt{a} ($a > 0$). Пусть $x^{(0)} = 1 > 0$. Воспользуемся формулой $x^{(i)} = \frac{1}{2} \left[x^{(i-1)} + \frac{a}{x^{(i-1)}} \right]$, получим итерационную последовательность $x^{(0)}, x^{(1)}, \dots, x^{(n)}, \dots$ $a = 2 : x^{(0)} = 1, x^{(1)} = 1.5, x^{(2)} = 1.41(6), x^{(3)} = 1.4142156\dots$. Известно, что метод сходится при любом начальном приближении $x^{(0)} > 0$.

Метод статистических испытаний основан на моделировании различных случайных величин и построении оценок. Этот метод применяют для решения не только тех задач, в которых в явном виде имеются случайные события, но также и для решения многих задач, не содержащих таких событий. В этом случае искусственно подбирают такое случайное явление, характеристики которого связаны с результатом решения исходной задачи. Для определения числовых значений этих характеристик и используется метод статистических испытаний.

Его главная идея основана на законе больших чисел, сам метод использует аппарат математической статистики.

Пример. Вычисление функций плотностей вероятностей различных законов, вычисление многомерных интегралов и тому подобные вопросы.

2. ПРИБЛИЖЕНИЕ ФУНКЦИЙ

2.1. Задача приближения функций

Вычисление значений функции $y = f(x)$ - задача, с которой постоянно приходиться сталкиваться на практике. Часто бывает, что вычисление $f(x)$ затруднительно, например:

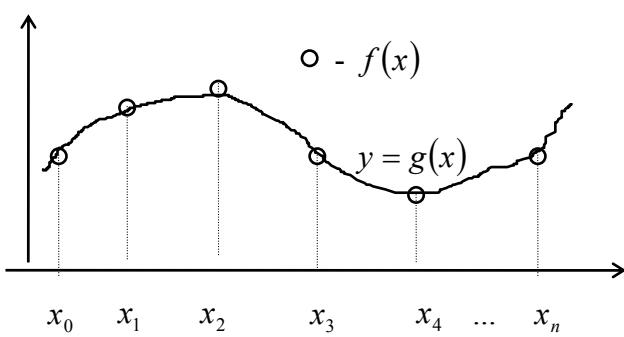
- 1) функция $y = f(x)$ задана таблично $y_i = f(x_i)$, $i = \overline{1, n}$, а вычисление необходимо проводить в точках x , не совпадающих с табличными;
- 2) вычисление функции $y = f(x)$ дорого;
- 3) для вычисления $y = f(x)$ необходим эксперимент.

В таких условиях целесообразно заменить $f(x)$ некоторой близкой к ней функцией $g(x)$, которая вычисляется быстро и надежно, а погрешность приближения $f(x) - g(x)$ достаточно мала. При этом полезно при выборе функции $g(x)$ использовать любую дополнительную информацию о функции $f(x)$, о ее гладкости, четности, периодичности, монотонности и так далее. Это дает возможность осознанно выбрать класс G аппроксимирующих функций.

Широко используются функции вида $\Phi_m(x) = a_0\varphi_0(x) + a_1\varphi_1(x) + \dots + a_m\varphi_m(x)$, представляющие собой линейные комбинации некоторых базисных функций $\varphi_0(x), \varphi_1(x), \dots, \varphi_m(x)$. Функция $\Phi_m(x)$ называется обобщенным многочленом степени m .

2.2. Интерполяция обобщенными многочленами

Если ставится требование совпадения функции $g(x)$ с функцией $f(x)$ в некоторых фиксированных точках, то это приводит к задаче интерполяции.



Построить функцию $g(x)$, удовлетворяющую условиям $g(x_i) = y_i$, $i = \overline{0, n}$. x_i - узлы интерполяции. Очевидно, что выбор $g(x)$ неоднозначен, так как по заданной таблице можно построить бесконечно много интерполирующих функций.

Рассмотрим обобщенный многочлен $\Phi_m(x)$, удовлетворяющий условию $\Phi_m(x_i) = y_i$, $i = \overline{0, n}$. Эта формула, представленная в виде $\Phi_m(x) = a_0\varphi_0(x) + a_1\varphi_1(x) + \dots + a_m\varphi_m(x)$, очевидно, эквивалентна следующей системе линейных алгебраических уравнений:

$$\left\{ \begin{array}{l} \varphi_0(x_0)a_0 + \varphi_1(x_0)a_1 + \dots + \varphi_m(x_0)a_m = y_0, \\ \varphi_0(x_1)a_0 + \varphi_1(x_1)a_1 + \dots + \varphi_m(x_1)a_m = y_1, \\ \dots \\ \varphi_0(x_n)a_0 + \varphi_1(x_n)a_1 + \dots + \varphi_m(x_n)a_m = y_n. \end{array} \right. \quad (2.2.1)$$

Для определения $\Phi_m(x)$ необходимо решить систему (2.2.1) относительно a_0, a_1, \dots, a_m . На практике это делается чрезвычайно редко. Как правило, система (2.2.1) плохо обусловле-

на. В большинстве приложений используются специальные явные формулы для записи $\Phi_m(x)$ и вычисление $\{a_m\}$ не нужно.

2.3. Полиномиальная интерполяция. Многочлен Лагранжа

Если в качестве базисной взять систему степенных функций, то есть $\{\varphi_0, \varphi_1, \dots, \varphi_n\} \rightarrow \{1, x, x^2, \dots, x^n\}$, то получаем задачу полиномиальной интерполяции:

$$P_n(x) = \sum_{k=0}^n a_k x^k, \quad P_n(x_i) = y_i, \quad i = \overline{0, n}. \quad (2.3.1)$$

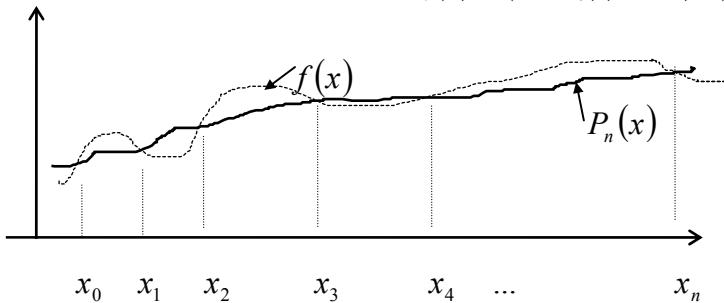
Теорема 2.1. Существует единственный интерполяционный многочлен степени n , удовлетворяющий условиям (2.3.1).

В качестве искомого многочлена возьмем многочлен степени n вида

$$P_n(x) = a_0(x - x_1)(x - x_2)\dots(x - x_n) + a_1(x - x_0)(x - x_2)\dots(x - x_n) + \dots + a_{n-1}(x - x_0)(x - x_1)(x - x_3)\dots(x - x_n) + \dots + a_n(x - x_0)(x - x_1)\dots(x - x_{n-1}). \quad (2.3.2)$$

Таким образом, система функций, по которой строится интерполяционный многочлен, есть

$$\begin{aligned}\varphi_0(x) &= (x - x_1)(x - x_2) \dots (x - x_n), \\ \varphi_1(x) &= (x - x_0)(x - x_2) \dots (x - x_n), \\ &\dots \\ \varphi_n(x) &= (x - x_0)(x - x_1) \dots (x - x_{n-1}).\end{aligned}$$



Для нахождения $P_n(x)$ надо найти набор коэффициентов a_0, a_1, \dots, a_n . Не будем составлять и решать систему линейных уравнений вида (2.2.1), найдем коэффициенты иным способом.

Пусть $x = x_0$, с учетом $P_n(x_0) = y_0 = a_0(x_0 - x_1)(x_0 - x_2)\dots(x_0 - x_n)$, получим

$$a_0 = \frac{y_0}{(x_0 - x_1)(x_0 - x_2) \dots (x_0 - x_n)}.$$

Аналогично, полагая $x = x_1$ и учитывая, что $y_1 = a_1(x_1 - x_0)(x_1 - x_2)\dots(x_1 - x_n)$, будем иметь

$$a_1 = \frac{y_1}{(x_1 - x_0)(x_1 - x_2) \dots (x_1 - x_n)}.$$

Если $x = x_n$, то $a_n = \frac{y_n}{(x_n - x_0)(x_n - x_1) \dots (x_n - x_{n-1})}$. Тогда сам многочлен $P_n(x)$ будет

иметь вид

$$\begin{aligned} P_n(x) &= a_0\varphi_0(x) + a_1\varphi_1(x) + \dots + a_n\varphi_n(x) = \\ &= \frac{(x-x_1)(x-x_2)\dots(x-x_n)}{(x_0-x_1)(x_0-x_2)\dots(x_0-x_n)}y_0 + \frac{(x-x_0)(x-x_2)\dots(x-x_n)}{(x_1-x_0)(x_1-x_2)\dots(x_1-x_n)}y_1 + \\ &+ \frac{(x-x_0)(x-x_1)\dots(x-x_n)}{(x_2-x_0)(x_2-x_1)\dots(x_2-x_n)}y_2 + \dots + \frac{(x-x_0)(x-x_1)\dots(x-x_{n-1})}{(x_n-x_0)(x_n-x_1)\dots(x_n-x_{n-1})}y_n. \end{aligned} \quad (2.3.3)$$

Эта формула называется интерполяционной формулой Лагранжа. Приведем ее в сокращенной записи:

$$P_n(x) = L_n(x) = \sum_{j=0}^n y_j l_{n_j}(x),$$

$$\varphi_{n_j} = l_{n_j}(x) = \prod_{\substack{k=0, \\ k \neq j}}^n \frac{x - x_k}{x_j - x_k} = \frac{(x - x_0)(x - x_1) \dots (x - x_{j-1})(x - x_{j+1}) \dots (x - x_n)}{(x_j - x_0)(x_j - x_1) \dots (x_j - x_{j-1})(x_j - x_{j+1}) \dots (x_j - x_n)}. \quad (2.3.4)$$

Очевидно, $l_{n_j}(x)$ представляет собой многочлен степени n , удовлетворяющий условию

$$l_{n_j}(x_i) = \begin{cases} 1, & i = j, \\ 0, & i \neq j. \end{cases}$$

Таким образом, степень многочлена $L_n(x)$ равна n , при $x = x_i$ в формуле (2.3.4) обращаются в нуль все слагаемые, кроме слагаемого с номером $j = i$, равного y_j .

Выпишем отдельно многочлены Лагранжа первой и второй степени, ибо именно они чаще всего используются на практике.

$$L_1(x) = y_0 \frac{x - x_1}{x_0 - x_1} + y_1 \frac{x - x_0}{x_1 - x_0} \text{ — прямая,} \quad (2.3.5)$$

$$L_2(x) = y_0 \frac{(x - x_1)(x - x_2)}{(x_0 - x_1)(x_0 - x_2)} + y_1 \frac{(x - x_0)(x - x_2)}{(x_1 - x_0)(x_1 - x_2)} + y_2 \frac{(x - x_0)(x - x_1)}{(x_2 - x_0)(x_2 - x_1)} \text{ — парабола.}$$

Пример. Написать интерполяционный многочлен Лагранжа для функции $f(x)$, значения которой заданы таблицей

i	0	1	2	3
x_i	0	0.1	0.3	0.5
y_i	-0.5	0	0.2	1.0

В данном случае $n = 3$, получаем при интерполяции кубическую параболу. Вычислим вначале $l_{n_j}(x)$:

$$l_{3,0} = \frac{(x - x_1)(x - x_2)(x - x_3)}{(x_0 - x_1)(x_0 - x_2)(x_0 - x_3)} = \frac{(x - 0.1)(x - 0.3)(x - 0.5)}{(0 - 0.1)(0 - 0.3)(0 - 0.5)} = \frac{x^3 - 0.9x^2 + 0.23x - 0.015}{0.015}.$$

$l_{3,1} \neq 0$, но его значение не понадобится, так как $y_1 = 0$. Не будем его вычислять.

$$l_{3,2} = \frac{(x - 0)(x - 0.1)(x - 0.5)}{(0.3 - 0)(0.3 - 0.1)(0.3 - 0.5)} = -\frac{x^3 - 0.6x^2 + 0.05x}{0.012},$$

$$l_{3,3} = \frac{(x - 0)(x - 0.1)(x - 0.3)}{(0.5 - 0)(0.5 - 0.1)(0.5 - 0.3)} = \frac{x^3 - 0.4x^2 + 0.03x}{0.04}.$$

Тогда искомый интерполяционный многочлен Лагранжа третьей степени будет выглядеть так $L_3(x) = l_{3,0}y_0 + l_{3,1}y_1 + l_{3,2}y_2 + l_{3,3}y_3 = 41.67x^3 - 30.00x^2 + 7.58x - 0.50$.

2.4. Погрешность интерполяции

Теорема 2.2. Пусть функция $f(x)$ дифференцируема $n+1$ раз на отрезке $[a, b]$, содержащем узлы интерполяции x_i , $i = \overline{0, n}$. Тогда для погрешности интерполяции в точке $x \in [a, b]$ справедливо равенство $f(x) - P_n(x) = \frac{f^{(n+1)}(\zeta)}{(n+1)!} \omega_{n+1}(x)$, в котором $\omega_{n+1}(x) = (x - x_0)(x - x_1)(x - x_2) \dots (x - x_n)$, а $\zeta \in (a, b)$.

Последнюю формулу несколько модернизируют. Так как положение точки ζ неизвестно, то $f^{(n+1)}(\zeta)$ заменяют на $M_{n+1} = \max_{[a, b]} |f^{(n+1)}(x)|$. Тогда

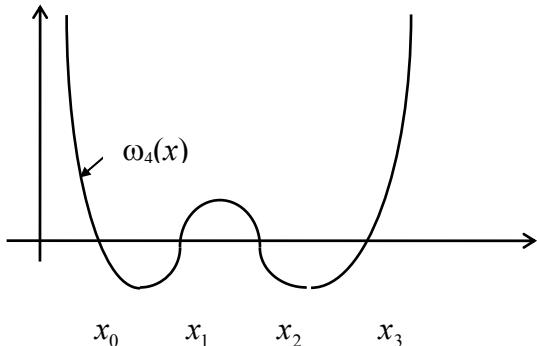
$$|f(x) - P_n(x)| \leq \frac{M_{n+1}}{(n+1)!} |\omega_{n+1}(x)| \text{ или } \max_{[a,b]} |f(x) - P_n(x)| \leq \frac{M_{n+1}}{(n+1)!} |\omega_{n+1}(x)|, M_{n+1} = \max_{[a,b]} |f^{(n+1)}(x)|,$$

$$\omega_{n+1}(x) = (x - x_0)(x - x_1)(x - x_2) \dots (x - x_n).$$

Пример. Вычислим значение $L_3(0.2)$ в предыдущем примере и оценим точность полученного значения:

$$L_3(0.2) = 41.67 \cdot 0.008 - 30.00 \cdot 0.04 + 7.58 \cdot 0.2 - 0.50 = 0.033 - 1.200 + 1.516 - 0.60 = -0.15.$$

$$|f(x) - L_3(x)| \leq \frac{M_4}{1 \cdot 2 \cdot 3 \cdot 4} |(x - x_0)(x - x_1)(x - x_2)(x - x_3)|.$$



Представление о типичном характере функции $\omega_{n+1}(x)$ дает график слева. При выходе x за пределы значений аргумента $\omega_{n+1}(x)$ быстро стремится к плюс бесконечности. Несколько огрубляя оценку погрешности, можно получить

$$\max_{[x_0, x_n]} |f(x) - L_n(x)| \leq \frac{M_{n+1}}{4(n+1)} h_{\max}^{n+1}, \text{ где } h_{\max} = \max_{1 \leq i \leq n} h_i. \text{ В}$$

нашем случае $h_{\max} = 0.2$ и

$$\max_{[-0.5, 1.0]} |f(x) - L_3(x)| \leq \frac{M_4}{16} 0.2^4 = 0.0001 \cdot M_4. \text{ Сколь-}$$

нибудь достоверную оценку $M_4 = \max_{[x_0, x_n]} |f''''(x)|$ здесь получить невозможно. Если предположить $M_4 < 1$, то $\varepsilon \approx 10^{-4}$.

2.5. Конечные разности и их свойства

Пусть функция $y = f(x)$ задана таблично $y_i = f(x_i)$, $i = \overline{0, n}$, $h = x_i - x_{i-1} = \text{const}$ - шаг таблицы, $x_0 < x_1 < \dots < x_n$ - узлы таблицы.

Величина $\Delta y_i = y_{i+1} - y_i$ **называется конечной разностью первого порядка** функции $y = f(x)$ в точке x_i с шагом h .

Конечная разность порядка k функции $y = f(x)$ в точке x_i есть $\Delta^k y_i = \Delta^{k-1} y_{i+1} - \Delta^{k-1} y_i$. **Таким образом, конечная разность второго порядка есть** $\Delta^2 y_i = \Delta y_{i+1} - \Delta y_i = y_{i+2} - y_{i+1} - y_{i+1} + y_i = y_{i+2} - 2y_{i+1} + y_i$. Аналогичным образом могут быть определены конечные разности произвольного порядка.

Конечные разности чаще всего располагают в виде таблицы следующим образом:

x	y	Δy	$\Delta^2 y$	$\Delta^3 y$...	$\Delta^n y$
x_0	y_0	Δy_0				
x_1	y_1	Δy_1	$\Delta^2 y_0$	$\Delta^3 y_0$		
x_2	y_2	Δy_2	$\Delta^2 y_1$			
x_3	y_3					$\Delta^n y_0$
...	...					
x_{n-1}	y_{n-1}	Δy_{n-2}	$\Delta^2 y_{n-2}$	$\Delta^3 y_{n-3}$		
x_n	y_n	Δy_{n-1}				

Теорема 2.3. *k*-я конечная разность выражается через значения функции в *k+1* точке по формуле $\Delta^k y_i = \sum_{j=0}^k (-1)^{k-j} C_k^j y_{i+j}$, где $C_k^j = \frac{k!}{j!(k-j)!}$. (2.5.1)

В частности уже получена $\Delta^2 y_i = y_{i+2} - 2y_{i+1} + y_i$, аналогично получаются формулы

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

$$\Delta^4 y_i = y_{i+4} - 4y_{i+3} + 6y_{i+2} - 4y_{i+1} + y_i.$$

Коэффициенты, входящие в эти формулы, можно взять из треугольника Паскаля*.

Теорема 2.4. Пусть функция $y = f(x)$ дифференцируема *k* раз на отрезке $[x_i, x_{i+k}]$.

Тогда справедливо равенство $\Delta^k y_i = h^k f^{(k)}(\zeta)$, $\zeta \in (x_i, x_{i+1})$. (2.5.2)

Доказательство

Теорема в общем виде доказывается по индукции. Проверим ее выполнимость только для $n = 1$.

$$\Delta f_k = hf'(\zeta) = f_{k+1} - f_k, \text{ то есть } f(x_k + h_k) = f(x_k) + h_k f'(\zeta), \zeta \in (x_k, x_{k+1}).$$

Но последняя формула - формула Лагранжа для $y = f(x)$. Для последующих *n* теорем доказывается по индукции. Эта формула может быть применима для оценки погрешности при интерполяции, когда функция задана только таблично. Если h_k мало, то $\left| \frac{\Delta^k f_0}{h^k} \right|$ можно приближенно принять за $M_k = \max_{[x_0, x_k]} |f^{(k)}(x)|$ и, таким образом, оценить погрешность $\Delta f(x^*)$.

В реальных вычислениях таблица конечных разностей $\Delta^k y_i$ строится по значениям y_i^* , каждое из которых содержит погрешность $\varepsilon_j = y_j - y_j^*$. Тогда в силу формулы (2.5.1) вычисленные значения $\Delta^k y_j^*$ содержат неустранимые ошибки

$$\varepsilon_i^{(k)} = \Delta^k y_i - \Delta^k y_i^* = \sum_{j=0}^k (-1)^{k-j} C_k^j \varepsilon_{i+j}. \quad (2.5.3)$$

Пусть $|\varepsilon_i| \leq \varepsilon$ для всех i , тогда можно получить гарантированную оценку

$$|\varepsilon_i^{(k)}| \leq \sum_{j=0}^k C_k^j \varepsilon = \varepsilon \sum_{j=0}^k C_k^j = 2^k \varepsilon. \quad (2.5.4)$$

2.6. Разделенные разности и их свойства

Пусть функция $y = f(x)$ задана на таблице $x_0, x_1, x_2, \dots, x_n$ значений аргумента с произвольным шагом, причем точки таблицы занумерованы также в произвольном порядке.

Величины $f(x_i; x_{i+1}) = \frac{f(x_{i+1}) - f(x_i)}{x_{i+1} - x_i}$ называются **разделенными разностями первого порядка** функции $y = f(x)$ в узлах x_i, x_{i+1} . Аналогично определяются **разделенные разности более высокого порядка**: $f(x_i; x_{i+1}; x_{i+2}) = \frac{f(x_{i+1}; x_{i+2}) - f(x_i; x_{i+1})}{x_{i+2} - x_i}$ - **разделенная разность второго порядка** в узлах x_i, x_{i+1}, x_{i+2} . Разделенной разностью *k*-го порядка называется число

$$f(x_i; x_{i+1}; \dots; x_{i+k}) = \frac{f(x_{i+1}; x_{i+2}; \dots; x_{i+k}) - f(x_i; x_{i+1}; \dots; x_{i+k-1})}{x_{i+k} - x_i}. \quad (2.6.1)$$

* Блез Паскаль (1623-1662) - французский математик.

Эти разности также можно записывать в виде треугольной таблицы:

x_0	$f(x_0)$				
		$f(x_0; x_1)$			
x_1	$f(x_1)$		$f(x_0; x_1; x_2)$		
		$f(x_1; x_2)$		$f(x_0; x_1; x_2; x_3)$	
x_2	$f(x_2)$		$f(x_1; x_2; x_3)$		$f(x_0; x_1; x_2; x_3; x_4)$
		$f(x_2; x_3)$		$f(x_1; x_2; x_3; x_4)$	
x_3	$f(x_3)$		$f(x_2; x_3; x_4)$		$f(x_1; x_2; x_3; x_4; x_5)$
		$f(x_3; x_4)$		$f(x_2; x_3; x_4; x_5)$	
x_4	$f(x_4)$		$f(x_3; x_4; x_5)$		
		$f(x_4; x_5)$			
x_5	$f(x_5)$				

Разделенные разности обладают рядом замечательных свойств, изложенных в следующих теоремах.

Теорема 2.5. Разделенная разность $f(x_i; x_{i+1}; \dots; x_{i+k})$ является симметричной функцией своих аргументов $x_i, x_{i+1}, \dots, x_{i+k}$ (то есть ее свойства не меняются при любой их перестановке).

Теорема 2.6. Разделенная разность k -го порядка выражается через значения функции следующим образом

$$f(x_0; x_1; \dots; x_k) = \sum_{i=0}^k \frac{f(x_i)}{(x_i - x_0)(x_i - x_1) \dots (x_i - x_{i-1})(x_i - x_{i+1}) \dots (x_i - x_k)}. \quad (2.6.2)$$

Легко заметить, что под знаком суммы стоят коэффициенты a_0, a_1, \dots, a_k обобщенного многочлена $\Phi_m(x)$, которые мы получали при выводе формулы Лагранжа (2.3.3). Теорема 2.6 доказывается методом математической индукции; проверим ее лишь для $k = 1$.

$$f(x_0; x_1) = \underbrace{\frac{f(x_0)}{x_0 - x_1} + \frac{f(x_1)}{x_1 - x_0}}_{\text{по формуле (2.6.2)}} = \underbrace{\frac{f(x_1) - f(x_0)}{x_1 - x_0}}_{\text{после приведения к общему знаменателю}}.$$

Теорема 2.7. Пусть функция $y = f(x)$ имеет на отрезке $[a, b]$, содержащем точки $x_i, x_{i+1}, \dots, x_{i+k}$, производную порядка k . Тогда справедливо равенство

$$f(x_i; x_{i+1}; \dots; x_{i+k}) = \frac{f^{(k)}(\zeta)}{k!}, \quad \zeta \in (a, b). \quad (2.6.3)$$

Теорема 2.8. В случае когда таблица значений аргумента имеет постоянный шаг h , конечная и разделенная разность связаны соотношением

$$f(x_i; x_{i+1}; \dots; x_{i+k}) = \frac{\Delta^k y_i}{h^k k!}. \quad (2.6.4)$$

Для $k = 1$ доказательство теоремы очевидно.

2.7. Интерполяционный многочлен Ньютона*

Пусть функция $y = f(x)$ задана в $n + 1$ точках таблично, то есть известны

* Исаак Ньютон (1643-1727) - английский физик, астроном и математик.

x_0	x_1	x_2	...	x_n
y_0	y_1	y_2	...	y_n

Алгебраический многочлен n -й степени

$$\begin{aligned} P_n(x) &= f(x_0) + f(x_0; x_1)(x - x_0) + f(x_0; x_1; x_2)(x - x_0)(x - x_1) + \dots \\ &+ f(x_0; x_1; \dots; x_n)(x - x_0)(x - x_1) \dots (x - x_{n-1}) = \sum_{k=0}^n f(x_0; x_1; \dots; x_k) \omega_k(x), \\ \omega_k(x) &= (x - x_0)(x - x_1) \dots (x - x_{k-1}), \quad \omega_0(x) \equiv 1 \end{aligned} \quad (2.7.1)$$

называется интерполяционным многочленом Ньютона с разделенными разностями.

Очевидна аналогия формулы (2.7.1) с формулой Тейлора. Действительно, так как по теореме 2.7 $f(x_0; x_1; \dots; x_{i+k}) = \frac{f^{(k)}(\zeta)}{k!}$, то $P_n(x) = f(x_0) + \frac{f'(\zeta)}{1!}(x - x_0) + \frac{f''(\zeta)}{2!}(x - x_0)(x - x_1) + \dots$

Формулы подраздела 2.4 о погрешности интерполяции $f(x) - P_n(x) = \frac{f^{(n+1)}(\zeta)}{(n+1)!} \omega_{n+1}(x)$ в точке

$$x, \text{ не являющейся узловой, можно уточнить следующим образом: } f(x) - P_n(x) = f(x_0; x_1; \dots; x_n; x) \omega_{n+1}(x). \quad (2.7.2)$$

В практическом плане формула (2.7.1) обладает рядом преимуществ перед формулой Лагранжа. Если, например, по каким-либо причинам необходимо увеличить степень интерполяционного многочлена на единицу, добавив в таблицу еще один узел x_{n+1} , то при использовании формулы Лагранжа это приведет не только к увеличению числа слагаемых, но и к необходимости вычислять каждое из них заново. В то же время для вычисления $P_n(x)$ по формуле Ньютона (2.7.1) достаточно добавить к $P_n(x)$ лишь очередное слагаемое, так как $P_{n+1}(x) - P_n(x) = f(x_0; x_1; \dots; x_n; x_{n+1}) \omega_{n+1}(x)$. Если величина $|x_{n+1} - x|$ мала, а функция $y = f(x)$ достаточно гладкая, то справедлива оценка: $f(x_0; x_1; \dots; x_n; x) \approx f(x_0; x_1; \dots; x_n; x_{n+1})$, из которой, с учетом предыдущего равенства, следует, что $f(x) - P_n(x) \approx P_{n+1}(x) - P_n(x)$. Тогда величину

$$\varepsilon_n = |P_{n+1}(x) - P_n(x)| \quad (2.7.3)$$

можно использовать для практической оценки погрешности интерполяции.

2.8. Вычислительная схема Эйткена*

Согласно этой схеме интерполяционные многочлены любого вида вычисляются последовательно по формулам

$$\begin{aligned} P_{i,i+1}(x) &= \frac{1}{x_{i+1} - x_i} \begin{vmatrix} y_i & x_i - x \\ y_{i+1} & x_{i+1} - x \end{vmatrix}, \\ P_{i,i+1,i+2}(x) &= \frac{1}{x_{i+2} - x_i} \begin{vmatrix} P_{i,i+1}(x) & x_i - x \\ P_{i+1,i+2}(x) & x_{i+2} - x \end{vmatrix}, \\ P_{i,i+1,i+2,i+3}(x) &= \frac{1}{x_{i+3} - x_i} \begin{vmatrix} P_{i,i+1,i+2}(x) & x_i - x \\ P_{i+1,i+2,i+3}(x) & x_{i+3} - x \end{vmatrix} \end{aligned} \quad (2.8.1)$$

и так далее. Интерполяционный многочлен n -й степени, принимающий в точках x_0, x_1, \dots, x_n значения $y_i, i = \overline{0, n}$, запишется следующим образом:

* Александр Крег Эйткен (1895-1967) - английский математик.

$$P_{0,1,2,\dots,n}(x) = \frac{1}{x_n - x_0} \begin{vmatrix} P_{0,1,\dots,n-1}(x) & x_0 - x \\ P_{1,2,\dots,n}(x) & x_n - x \end{vmatrix}. \quad (2.8.2)$$

Действительно, из первой формулы (2.8.1) при $i = 0$ сразу получаем

$$\begin{aligned} P_{0,1}(x) &= \frac{y_0(x_1 - x) - y_1(x_0 - x)}{x_1 - x_0} = y_0 \frac{x - x_1}{x_0 - x_1} + y_1 \frac{x - x_0}{x_1 - x_0} \equiv L_1(x) = \\ &= y_0 \frac{x - x_0 + x_0 - x_1}{x_0 - x_1} + y_1 \frac{x - x_0}{x_1 - x_0} = y_0 \frac{x_0 - x_1}{x_0 - x_1} + y_0 \frac{x - x_0}{x_0 - x_1} + y_1 \frac{x - x_0}{x_1 - x_0} = \\ &= y_0 + \frac{(y_1 - y_0)(x - x_0)}{x_1 - x_0} = y_0 + \frac{y_1 - y_0}{x_1 - x_0} (x - x_0) = y_0 + f(x_0; x_1)(x - x_0) \equiv P_1(x). \end{aligned}$$

Остальные формулы проверяются аналогично. Кроме того, мы получили, что $L_1(x) \equiv P_1(x)$. Это действительно так по теореме о единственности интерполяционного многочлена n -й степени. Таким образом, $L_n(x)$ и $P_n(x)$ тождественно совпадают и являются по сути лишь разной формой записи единого интерполяционного многочлена n -й степени.

Схема Эйткена применяется там, где не нужно общее выражение $P_n(x)$, а нужно лишь его значение при конкретных x , и при этом значения функции даны в достаточно большом числе узлов. Вычисления по схеме Эйткена удобно вести с помощью таблицы, аналогичной таблице конечных или разделенных разностей:

x_0	y_0			
		$P_{0,1}$		
x_1	y_1		$P_{0,1,2}$	
		$P_{1,2}$		
x_2	y_2		$P_{1,2,3}$	
		$P_{2,3}$		
x_3	y_3			
\dots	\dots			$P_{n-3,n-2,n-1,n}$
x_{n-1}	y_{n-1}		$P_{n-2,n-1,n}$	
		$P_{n-1,n}$		
x_n	y_n			

Вычисления прекращают, если $|P_{n+1}(x) - P_n(x)| < \varepsilon$ или если последовательные значения $P_{0,1,\dots,n}(x)$ и $P_{0,1,\dots,n,n+1}(x)$ совпадут в пределах заданной точности.

Пример. Вычислить $y = \sqrt[3]{x}$ по схеме Эйткена в точке $x = 1.15$, если $y = f(x)$ задана таблицей:

x	1.0	1.1	1.3	1.5	1.6
y	1.000	1.032	1.091	1.145	1.170

Составим таблицу и заполним по формулам (2.8.1) ее столбцы, начиная с четвертого:

$$P_{0,1} = \frac{1}{x_1 - x_0} \begin{vmatrix} y_0 & x_0 - x \\ y_1 & x_1 - x \end{vmatrix} = \frac{1}{0.1} \begin{vmatrix} 1.000 & -0.15 \\ 1.032 & -0.05 \end{vmatrix} = 1.048,$$

$$P_{1,2} = \frac{1}{x_2 - x_1} \begin{vmatrix} y_1 & x_1 - x \\ y_2 & x_2 - x \end{vmatrix} = \frac{1}{0.2} \begin{vmatrix} 1.032 & -0.05 \\ 1.091 & 0.15 \end{vmatrix} = 1.047,$$

$$P_{2,3} = \frac{1}{x_3 - x_2} \begin{vmatrix} y_2 & x_2 - x \\ y_3 & x_3 - x \end{vmatrix} = \frac{1}{0.2} \begin{vmatrix} 1.091 & 0.15 \\ 1.145 & 0.35 \end{vmatrix} = 1.050,$$

$$P_{3,4} = \frac{1}{x_4 - x_3} \begin{vmatrix} y_3 & x_3 - x \\ y_4 & x_4 - x \end{vmatrix} = \frac{1}{0.1} \begin{vmatrix} 1.145 & 0.35 \\ 1.170 & 0.45 \end{vmatrix} = 1.057.$$

Следующий столбец таблицы заполняется аналогично:

$$P_{0,1,2} = \frac{1}{x_2 - x_0} \begin{vmatrix} P_{0,1} & x_0 - x \\ P_{1,2} & x_2 - x \end{vmatrix} = \frac{1}{0.3} \begin{vmatrix} 1.048 & -0.15 \\ 1.047 & 0.15 \end{vmatrix} = 1.048.$$

На этом вычисления можно прекратить, так как $P_{0,1}$ и $P_{0,1,2}$ совпадают до третьего знака, следовательно, $\sqrt[3]{1.15} = 1.048$ с точностью $\varepsilon = 10^{-3}$. Итоговая таблица с результатами вычислений приведена ниже.

i	x_i	y_i	$x_i - x$	$P_{i,i+1}$	$P_{i,i+1,i+2}$
0	1.0	1.000	-0.15		
				1.048	
1	1.1	1.032	-0.05		1.048
				1.047	
2	1.3	1.091	0.15		
				1.050	
3	1.5	1.145	0.35		
				1.057	
4	1.6	1.170	0.45		

2.9. Лабораторная работа № 2. Интерполярование и экстраполирование данных. Интерполяционный многочлен Лагранжа

В практических расчетах чаще всего используют интерполяционные многочлены небольших степеней. Из формулы (2.3.4) видно, что единственный, восстанавливаемый без дополнительных условий многочлен Лагранжа должен иметь степень, на единицу меньшую числа точек интерполяционной таблицы. Обычно по исходным данным именно такой многочлен и восстанавливается. По найденному многочлену находят приближенные значения функции для любых значений аргумента, лежащих между узлами заданной сетки.

Пример. Найти многочлен наименьшей степени, принимающий в данных точках заданные значения:

i	0	1	2
x	1.45	1.36	1.14
y	3.14	4.15	5.65

Здесь заданы лишь три узла сетки. Следовательно, можно восстановить многочлен Лагранжа второго порядка: $L_2(x) = \sum_{i=0}^2 y_i \cdot l_{2i}(x) = l_{20}y_0 + l_{21}y_1 + l_{22}y_2 =$

$= \frac{(x - x_1)(x - x_2)}{(x_0 - x_1)(x_0 - x_2)} y_0 + \frac{(x - x_0)(x - x_2)}{(x_1 - x_0)(x_1 - x_2)} y_1 + \frac{(x - x_0)(x - x_1)}{(x_2 - x_0)(x_2 - x_1)} y_2$. Вычислим вначале коэффициенты

$$l_{20}(x) = \frac{(x - 1.36)(x - 1.14)}{(1.45 - 1.36)(1.45 - 1.14)} = \frac{x^2 - 1.360x - 1.140x + 1.550}{0.090 \cdot 0.310} =$$

$$= \frac{x^2 - 2.500x + 1.550}{0.0279}, \quad l_{21}(x) = \frac{(x - 1.45)(x - 1.14)}{(1.36 - 1.45)(1.36 - 1.14)} = \frac{x^2 - 1.450x - 1.140x + 1.653}{-0.090 \cdot 0.220} =$$

$$= \frac{x^2 - 2.590x + 1.653}{-0.0198}, \quad l_{22}(x) = \frac{(x - 1.45)(x - 1.36)}{(1.14 - 1.45)(1.14 - 1.36)} = \frac{x^2 - 1.450x - 1.360x + 1.972}{-0.310 \cdot (-0.220)} =$$

$$= \frac{x^2 - 2.810x + 1.972}{0.0682}.$$

Тогда $L_2(x) = 112.5448 \cdot (x^2 - 2.500x + 1.550) - 209.5960 \cdot$

$$(x^2 - 2.590x + 1.653) + 82.8446 \cdot (x^2 - 2.810x + 1.972) = 112.5448x^2 - 281.3620x + 174.4444 - 209.5960x^2 + 542.8536x - 346.4622 + 82.8446x^2 - 232.7933x + 163.3696 = -14.2066x^2 + 28.6983x - 8.6483.$$

Для вычисления лагранжевых коэффициентов l_{ni} удобно применить следующую таблицу разностей (2.9.1):

$x - x_0$	$x_0 - x_1$	$x_0 - x_2$...	$x_0 - x_n$
$x_1 - x_0$	$x - x_1$	$x_1 - x_2$		$x_1 - x_n$
$x_2 - x_0$	$x_2 - x_1$	$x - x_2$...	$x_2 - x_n$
...
$x_n - x_0$	$x_n - x_1$	$x_n - x_2$...	$x - x_n$

на главной диагонали которой стоят разности вида $x - x_i$, $i = \overline{0, n}$. Если обозначить произведение элементов i -й строки через D_i , а произведение элементов главной диагонали через $P_{n+1}(x)$, то

$$l_{ni}(x) = \frac{P_{n+1}(x)}{D_i}, i = \overline{0, n}. \quad (2.9.2)$$

Для рассмотренного выше примера таблица разностей будет иметь вид

$x - x_0$	$x_0 - x_1$	$x_0 - x_2$
$x_1 - x_0$	$x - x_1$	$x_1 - x_2$
$x_2 - x_0$	$x_2 - x_1$	$x - x_2$

Тогда, например, для l_{30} получим $P_3(x) = (x - x_0)(x - x_1)(x - x_2)$, $D_0 = (x - x_0)(x_0 - x_1)(x_0 - x_2)$ и $l_{30}(x) = \frac{P_3(x)}{D_0} = \frac{(x - x_1)(x - x_2)}{(x_0 - x_1)(x_0 - x_2)}$, что совпадает с выражением, найденным по формуле (2.3.4).

Теперь по найденному интерполяционному многочлену Лагранжа можно посчитать приближенное значение функции $y = f(x)$ для любого значения аргумента $x \in [1.14, 1.45]$. Вычислим, например, $y(1.415)$: $y(1.415) \approx -14.2066 \cdot (1.415)^2 + 28.6983 \cdot (1.415) - 8.6483 = 3.5150$. Результат необходимо округлить до второго знака после запятой. Итак, $y(1.415) \approx 3.52$.

Если требуется найти не общее выражение $L_n(x)$, а лишь его значения при конкретных значениях аргумента, то удобно пользоваться интерполяционной схемой Эйткена по

формулам (2.8.1) и (2.8.2). Вычислим еще раз $y(1.415)$, расположив данные по схеме Эйткена в удобной таблице (2.9.3):

i	x_i	y_i	$x_i - x$	$L_{i,i+1}$	$L_{i,i+1,i+2}$
0	x_0	y_0	$x_0 - x$		
1	x_1	y_1	$x_1 - x$	$L_{0,1}$	
2	x_2	y_2	$x_2 - x$	$L_{1,2}$	$L_{0,1,2}$

Вычисления по схеме Эйткена обычно ведут до тех пор, пока последовательные значения $L_{0,1,\dots,n}$ и $L_{0,1,\dots,n,n+1}$ не совпадут в пределах заданной точности.

В нашем случае из-за малого количества узлов таблица получилась очень короткой, содержащей всего три многочлена: $L_{0,1}$, $L_{1,2}$ и $L_{0,1,2}$. Вычислим их:

$$L_{0,1} = \frac{1}{x_1 - x_0} \begin{vmatrix} y_0 & x_0 - x \\ y_1 & x_1 - x \end{vmatrix} = \frac{1}{-0.09} \begin{vmatrix} 3.14 & 0.035 \\ 4.15 & -0.055 \end{vmatrix} = 3.5328,$$

$$L_{1,2} = \frac{1}{x_2 - x_1} \begin{vmatrix} y_1 & x_1 - x \\ y_2 & x_2 - x \end{vmatrix} = \frac{1}{-0.22} \begin{vmatrix} 4.15 & -0.055 \\ 5.65 & -0.275 \end{vmatrix} = 3.7750,$$

$$L_{0,1,2} = \frac{1}{x_2 - x_0} \begin{vmatrix} L_{0,1} & x_0 - x \\ L_{1,2} & x_2 - x \end{vmatrix} = \frac{1}{-0.31} \begin{vmatrix} 3.5328 & 0.035 \\ 3.7750 & -0.275 \end{vmatrix} = 3.5601.$$

i	x_i	y_i	$x_i - x$	$L_{i,i+1}$	$L_{i,i+1,i+2}$
0	1.45	3.14	0.035		
1	1.36	4.15	-0.055	3.5328	
2	1.14	5.65	-0.275	3.7750	3.5601

Итак, $y(1.415) \approx L_{0,1,2} = 3.56$.

Приведенный пример показывает, что значение таблично заданной функции в точке лучше вычислять по схеме Эйткена, что обеспечивает большую точность при меньших вычислительных затратах. При определении лагранжевых коэффициентов l_{ni} возможна потеря точности из-за малых знаменателей соответствующих дробей, кроме того восстановленный $L_2(x)$ имеет большие и противоположные по знаку коэффициенты, что также требует сохранения лишних знаков в промежуточных вычислениях.

Задание № 1. По образцу разобранного примера восстановить многочлен Лагранжа, удовлетворяющий приведенным исходным данным.

1.

i	0	1	2	3
x	0	1	2	5
y	2	3	12	147

2.

i	0	1	2	3
x	-2	1	2	4
y	25	-8	-15	-23

3.

i	0	1	2	3	4
x	-2	-1	0	1	2
y	6	0	2	0	6

4.

i	0	1	2	3
x	0	1	2	5
y	3	4	13	148

5.

i	0	1	2	3
x	-2	1	2	4
y	26	-7	-14	-22

6.

i	0	1	2	3	4
x	-2	-1	0	1	2
y	5	0	1	0	5

7.

i	0	1	2	3
x	-1	0	1	4
y	2	3	12	147

8

i	0	1	2	3
x	1	2	3	6
y	2	3	12	147

9.

i	0	1	2	3
x	-3	0	1	3
y	25	-8	-15	-23

10.

i	0	1	2	3
x	-1	2	3	5
y	25	-8	-15	-23

11.

i	0	1	2	3	4
x	-3	-2	-1	0	4
y	6	0	2	0	6

12.

i	0	1	2	3	4
x	-1	0	1	2	3
y	6	0	2	0	6

13.

i	0	1	2	3
x	2	3	4	7
y	2	3	12	147

14.

i	0	1	2	3
x	-2	-1	0	3
y	2	3	12	147

15.

i	0	1	2	3
x	-4	-1	0	2
y	25	-8	-15	-23

16.

i	0	1	2	3
x	0	3	4	6
y	25	-8	-15	-23

17.

i	0	1	2	3
x	-1	0	1	4
y	3	4	13	148

18.

i	0	1	2	3
x	1	2	4	6
y	1	2	34	146

19.

i	0	1	2	3
x	-3	0	1	3
y	26	-7	-14	-22

20.

i	0	1	2	3
x	-1	2	3	5
y	26	-7	-14	-22

21.

i	0	1	2	3	4
x	-3	-2	-1	0	1
y	7	1	3	1	7

22.

i	0	1	2	3	4
x	-1	0	1	2	3
y	5	-1	1	-1	5

23.

i	0	1	2	3	4
x	-1	0	1	2	3
y	2	1	0	1	10

24.

i	0	1	2	3
x	-2	-1	0	1
y	1	6	5	4

25.

i	0	1	2	3
x	-3	-2	-1	0
y	40	27	12	1

26.

i	0	1	2	3	4
x	-2	-1	0	1	2
y	-27	-4	-1	-6	-7

27.

i	0	1	2	3
x	-1	0	1	2
y	-5	-10	-1	34

28.

i	0	1	2	3	4
x	-2	-1	0	1	2
y	16	-1	0	1	8

29.

i	0	1	2	3	4
x	-2	-1	0	1	2
y	-23	-6	1	-2	9

30.

i	0	1	2	3
x	1	2	3	4
y	1	2	13	40

Задание № 2. Используя схему Эйткена, вычислить приближенное значение функции $y = f(x)$, заданной таблично при данном значении аргумента x_T .

I.

№ варианта	x_T	x	y
1	1.020	0.20	10.2316
6	1.140	0.80	10.9590
11	1.785	1.20	11.4725
16	2.031	1.70	12.1483
21	3.255	2.35	13.0120
26	3.512	3.00	14.0976
		3.14	16.2134
		3.61	19.0218

II.

№ варианта	x_T	x	y
2	0.616	0.45	2.5742
7	0.478	0.47	2.3251
12	0.665	0.52	2.0934
17	0.537	0.61	1.8620
22	0.749	0.66	1.7493
27	0.528	0.70	1.6210
		0.74	1.3418
		0.79	1.1212

III.

№ варианта	x_T	x	y

3	3.140	1.11	9.0542
8	2.351	1.59	6.6166
13	3.329	2.10	4.6917
18	2.756	2.96	3.3511
23	1.316	3.54	2.7995
28	5.555	4.00	2.3652
		5.36	1.9418
		5.78	1.8291

IV.

№ варианта	x_T	x	y
4	0.702	0.43	1.6360
9	0.512	0.48	1.7323
14	0.645	0.55	1.8769
19	0.736	0.62	2.0335
24	0.608	0.70	2.2285
29	0.762	0.75	2.3597
		0.77	2.4103
		0.83	2.6978

V.

№ варианта	x_T	x	y
5	8.960	6.81	8.0866
10	7.675	7.35	8.9492
15	9.490	8.00	10.2964
20	10.000	8.85	12.0966
25	8.125	9.34	13.4087
30	10.777	9.95	15.2368
		10.25	18.1109
		11.00	21.5445

Выполним теперь те же действия в пакете Mathcad, освоив простейшие символьные вычисления, панель программирования, а также действия с векторами и матрицами. Рассмотрим и решим две задачи, аналогичные заданиям № 1 и № 2: построим многочлен Лагранжа по заданной таблице и вычислим приближенное значение функции по схеме Эйткена. Для выполнения обеих этих задач удобно воспользоваться аппаратом матричной алгебры пакета.

Кнопка  -Vector and Matrix Palette (Панель векторных и матричных опе-раций) открывает панель, изображенную чуть ниже, кнопки которой соответствуют



следующим операциям:

-  - определение размеров матрицы;
-  - численное вычисление определителя матрицы;
-  - поэлементные операции с матрицами (режим векторизации);
-  - скалярное произведение двух векторов;



- векторное произведение;



- вычисление суммы компонент вектора $\sum v = \sum_{i=1}^n v_i$;



- определение столбца матрицы: $M^{(j)}$ – j -й столбец матрицы;



- транспонирование матрицы;



- визуализация цифровой информации, сохраненной в матрице.

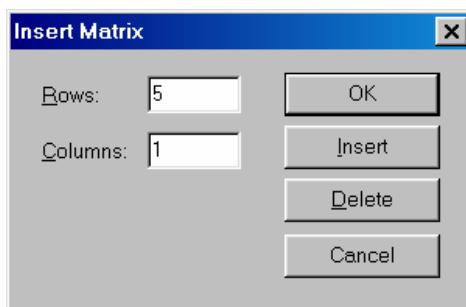
Три действия: транспонирование, обращение матрицы и вычисление определителя – можно выполнить и из меню символьных операций (Symbolics).

Имеется три способа создания векторов и матриц в пакете Mathcad: заполнением массива пустых полей, предварительно созданной матрицы нужной размерности с помощью панели определения размеров; использованием дискретного аргумента, чтобы определить элементы матрицы; считыванием матрицы из файлов данных.

Итак, рассмотрим первую задачу. Пусть функция $y = f(x)$ задана таблицей

x	-1	0	1	2	3
y	5	1	1	11	61

Введем исходные данные и матрицу, соответствующую таблице (2.9.1). Для того чтобы определить вектор или матрицу, введем с клавиатуры имя вектора и знак присваивания. Затем щелкнем по кнопке и откроем окно диалога определения размеров матрицы:



После этого заполним массив пустых полей, получим на экране монитора

$$ORIGIN := 1$$

$$x := \begin{pmatrix} -1 \\ 0 \\ 1 \\ 2 \\ 3 \end{pmatrix} \quad y := \begin{pmatrix} 5 \\ 1 \\ 1 \\ 11 \\ 61 \end{pmatrix}$$



Переменная ORIGIN определяет начало нумерации строк и столбцов матрицы в пакете Mathcad, она описана в предыдущей лабораторной работе. Для определения таблицы (2.9.1) нужно задать функциональную, а не числовую матрицу, поэтому набираем

$$M1(x) := \begin{pmatrix} x+1 & -1 & -2 & -3 & -4 \\ 1 & x & -1 & -2 & -3 \\ 2 & 1 & x-1 & -1 & -2 \\ 3 & 2 & 1 & x-2 & -1 \\ 4 & 3 & 2 & 1 & x-3 \end{pmatrix}$$

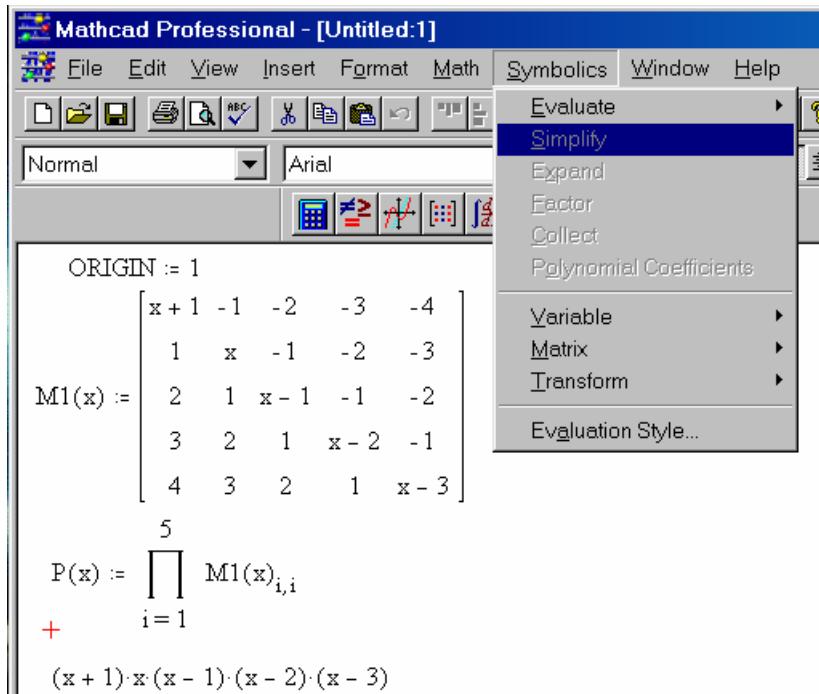


Далее вычисляем ла-

гравежевы коэффициенты по формуле (2.9.2). Для этого необходимо преобразовать символьное выражение. Кнопка - Calculus Palette (Панель операций математического анализа) открывает панель, с помощью которой можно суммировать, дифференцировать, интегрировать, вычислять произведения и находить пределы как числовых, так и символьных выражений. Введем выражение

$$P(x) := \prod_{i=1}^5 M1(x)_{i,i}$$

Щелчком по строке Simplify (Упростить) меню Symbolics (Символьные вычисления) символьному процессору передается выражение, выделенное в рабочем документе, а преобразованное выражение отображается в рабочем документе ниже или справа от исходного выражения в зависимости от задания стиля выражения (Evaluation Style).



В нашем случае получим на экране

$$P(x) := \prod_{i=1}^5 M1(x)_{i,i}$$

$$(x + 1) \cdot x \cdot (x - 1) \cdot (x - 2) \cdot (x - 3)$$

Аналогично вычисляются знаменатели формулы (2.9.2):

$$D1(x) := \prod_{i=1}^5 M1(x)_{1,i} \quad D2(x) := \prod_{i=1}^5 M1(x)_{2,i} \quad D3(x) := \prod_{i=1}^5 M1(x)_{3,i}$$

$$\frac{24 \cdot x + 24}{-6 \cdot x} \quad \frac{-6 \cdot x}{4 \cdot x - 4}$$

$$D4(x) := \prod_{i=1}^5 M1(x)_{4,i} \quad D5(x) := \prod_{i=1}^5 M1(x)_{5,i}$$

$$\frac{-6 \cdot x + 12}{24 \cdot x - 72}$$

Далее таким же образом вычисляем лагранжевы коэффициенты:

$$l1(x) := \frac{P(x)}{D1(x)} \quad l2(x) := \frac{P(x)}{D2(x)}$$

$$\frac{1}{24} \cdot x(x - 1) \cdot (x - 2) \cdot (x - 3) \quad - \frac{1}{6} \cdot (x + 1) \cdot (x - 1) \cdot (x - 2) \cdot (x - 3)$$

$$l3 := \frac{P(x)}{D3(x)}$$

$$l4 := \frac{P(x)}{D4(x)}$$

$$\frac{1}{4} \cdot (x+1) \cdot x \cdot (x-2) \cdot (x-3) - \frac{1}{6} \cdot (x+1) \cdot x \cdot (x-1) \cdot (x-3)$$

$$l5(x) := \frac{P(x)}{D5(x)}$$

$$\frac{1}{24} \cdot (x+1) \cdot x \cdot (x-1) \cdot (x-2)$$

Наконец, получим сам многочлен Лагранжа:

$$\text{Lag}(x) := y_1 \cdot l1(x) + y_2 \cdot l2(x) + y_3 \cdot l3(x) + y_4 \cdot l4(x) + y_5 \cdot l5(x)$$

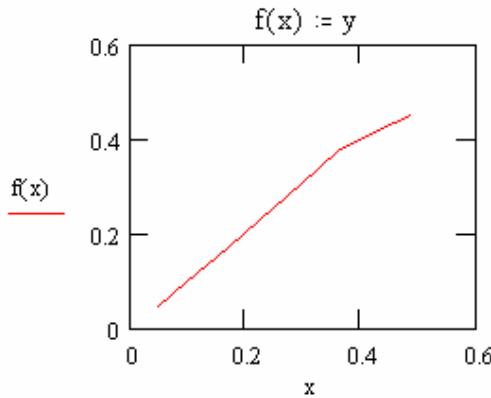
$$x^4 - x^3 + x^2 - x + 1$$

По найденной формуле можно приближенно рассчитывать значения функции в любых точках как внутри, так и вне табличного интервала $[-1, 3]$ с точностью, определяемой теоремой 2.2 (подразд. 2.4), например, $\text{Lag}(4) = 205$.

Решим теперь задачу задания № 2. Пусть $y = f(x)$ задана таблицей и $x_T = 0.2994$.

x	0.05	0.10	0.17	0.25	0.30	0.36	0.43	0.49
y	0.0500	0.1003	0.1717	0.2553	0.3093	0.3767	0.4218	0.4536

Построим сначала, как описано в предыдущей лабораторной работе, график этой зависимости:



Далее последовательно строим числовую матрицу (2.9.3) схемы Эйткена при помощи дискретного аргумента:

$$i := 1 \dots 8 \quad \text{Lag}_{i,1} := x_i \quad \text{Lag}_{i,2} := y_i \quad \text{Lag}_{i,3} := x_i - xt$$

$$i := 1 \dots 7 \quad \text{Lag}_{i+1,4} := \frac{1}{x_{i+1} - x_i} \cdot (y_i \cdot \text{Lag}_{i+1,3} - y_{i+1} \cdot \text{Lag}_{i,3})$$

$$\text{Lag} = \begin{bmatrix} 0.05 & 0.05 & -0.2494 & 0 \\ 0.1 & 0.1003 & -0.1994 & 0.3009 \\ 0.17 & 0.1717 & -0.1294 & 0.30369 \\ 0.25 & 0.2553 & -0.0494 & 0.30692 \\ 0.3 & 0.3093 & 6 \cdot 10^{-4} & 0.30865 \\ 0.36 & 0.3764 & 0.0606 & 0.30863 \\ 0.43 & 0.4218 & 0.1306 & 0.3371 \\ 0.49 & 0.4536 & 0.1906 & 0.35258 \end{bmatrix}$$

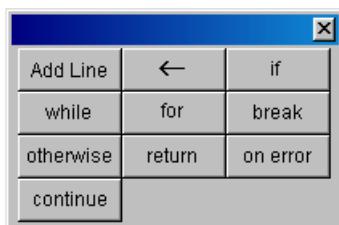
$$\begin{aligned}
i := 1 \dots 6 \quad & \text{Lag}_{i+2,5} := \frac{1}{x_{i+2} - x_i} \cdot (\text{Lag}_{i+2,3} \cdot \text{Lag}_{i+1,4} - \text{Lag}_{i,3} \cdot \text{Lag}_{i+2,4}) \\
i := 1 \dots 5 \quad & \text{Lag}_{i+3,6} := \frac{1}{x_{i+3} - x_i} \cdot (\text{Lag}_{i+3,3} \cdot \text{Lag}_{i+2,5} - \text{Lag}_{i,3} \cdot \text{Lag}_{i+3,5}) \\
i := 1 \dots 4 \quad & \text{Lag}_{i+4,7} := \frac{1}{x_{i+4} - x_i} \cdot (\text{Lag}_{i+4,3} \cdot \text{Lag}_{i+3,6} - \text{Lag}_{i,3} \cdot \text{Lag}_{i+4,6}) \\
i := 1 \dots 3 \quad & \text{Lag}_{i+5,8} := \frac{1}{x_{i+5} - x_i} \cdot (\text{Lag}_{i+5,3} \cdot \text{Lag}_{i+4,7} - \text{Lag}_{i,3} \cdot \text{Lag}_{i+5,7})
\end{aligned}$$

$\text{Lag} = \left[\begin{array}{ccccccccc} 0.05 & 0.05 & -0.2494 & 0 & 0 & 0 & 0 & 0 \\ 0.1 & 0.1003 & -0.1994 & 0.3009 & 0 & 0 & 0 & 0 \\ 0.17 & 0.1717 & -0.1294 & 0.30369 & 0.3067 & 0 & 0 & 0 \\ 0.25 & 0.2553 & -0.0494 & 0.30692 & 0.30799 & 0.30831 & 0 & 0 \\ 0.3 & 0.3093 & 6 \cdot 10^{-4} & 0.30865 & 0.30864 & 0.30864 & 0.30864 & 0 \\ 0.36 & 0.3764 & 0.0606 & 0.30863 & 0.30864 & 0.30864 & 0.30864 & 0.30864 \\ 0.43 & 0.4218 & 0.1306 & 0.3371 & 0.3085 & 0.3086 & 0.30862 & 0.30863 \\ 0.49 & 0.4536 & 0.1906 & 0.35258 & 0.32988 & 0.30843 & 0.30857 & 0.3086 \end{array} \right]$

Во всех приведенных таблицах задано пять знаков после запятой в данных, выводимых на дисплей. Это сделано таким образом. После двойного щелчка левой мыши открывается окно ввода параметров настройки отображения чисел, затем выставляется цифра 5 в верхнем окошке.

К сожалению, записать двойной цикл и вычислить все в одну строку с помощью дискретного аргумента здесь не удастся, так как, во-первых, нельзя в пакете Mathcad определять простую переменную через дискретный аргумент, во-вторых, переменные, определяющие дискретный диапазон, также не могут определяться через предыдущий дискретный аргумент. Однако эту задачу можно легко решить, составив коротенькую программу вычисления таблицы схемы Эйткена.

Воспользуемся кнопкой  -Programming Palette (Панель программи-рования). Она открывает следующую панель:



Кнопки этой панели вызывают операторы, во многом аналогичные основным операторам распространенных языков программирования. Язык Mathcad очень напоминает черты языка С. В этом нет ничего удивительного, так как сам Mathcad написан на этом языке. Основными составными операторами являются операторы выбора (**if**), повторения (**until**) и цикла (**for**) с заранее известным числом повторений. Кроме того, введено понятие локальной переменной, операторы досрочного выхода из цикла (**break**) и (**continue**) и оператор досрочного выхода из программы (**return**).

Кнопка  - это команда добавления строки в программу или в тело цикла.

С помощью этой кнопки алгоритмические конструкции в программе или цикле могут быть сделаны сколь угодно большими. При нажатии этой кнопки в окне рабочего документа появляется  конструкция вида  . Вертикальная линия объединяет отдельные

операторы в операторный блок с одним входом и одним выходом, выполняющийся как единый оператор. Кнопка -это оператор присваивания значения локальной переменной.

Кнопка вводит в программы альтернативу с одним плечом

(неполная альтернатива). В первый квадратик (левее **if**) записывается выполняемый оператор, в правый - булево выражение. Выполняемый оператор исполнится, если булево выражение возвращает значение «Да» (в среде Mathcad – любое числовое значение, отличное от нуля). Для организации полной альтернативы используется кнопка с оператором **otherwise** . Кнопка -это кнопка цикла с проверкой.

while В правый квадратик записывается булево выражение, управляющее цик-

лом, в нижний – тело цикла. Кнопка вводит в программы цикл с параметром.

for

В первый квадрат вводится переменная цикла, в правом (после знака \in) указываются пределы изменения этого параметра. Эти пределы можно просто перечислять через запятую или указывать в том виде, как для дискретной переменной: $t_1, t_1 + \Delta t, t_k$. В нижний квадрат помещается тело цикла.

Кнопки и позволяют досрочно выходить из

циклов **while** и **for**, а кнопка - из всей программы. При этом оператор **break** передает управление обязательно в конец цикла, а оператор **continue** - в любое место цикла.

Наконец, кнопка вводит оператор обработки ошибок. Если при выпол-

on error

нении оператора в правом квадрате возникнет ошибка, то будет выполняться оператор из левого квадрата. Оператор **on error** полезен в тех случаях, когда ошибку сложно локализовать, тем более что в программах пакета Mathcad, к сожалению, невозможно выводить на дисплей промежуточные результаты расчета.

Итак, программа расчета матрицы схемы Эйткена может быть, например, такой:

```

Lag(x,y,z,n) := | M<1> ← x
                  | M<2> ← z
                  | M<3> ← y
                  | for i ∈ 1..n - 1
                  |   for j ∈ 1..n - i
                  |     a ← 1 / (xi+j - xj)
                  |     det ← Mi+j,2 · Mi+j-1,i+2 - Mj,2 · Mi+j,i+2
                  |     Mi+j,i+3 ← a · det
                  |
M

```

Параметрами программы являются вектора исходных данных x и y . Компоненты вектора z равны: $z_i = x_i - x_T$, n - число точек табличного задания функции $y = f(x)$. Для получения матрицы по схеме Эйткена необходимо определить лишь вектор z и саму матри-

$$i := 1..8 \quad z_1 := x_1 - xt \quad MM := Lag(x, y, z, 8)$$

	1	2	3	4	5	6	7	8	9	10
1	0.05	-0.2494	0.05	0	0	0	0	0	0	0
2	0.1	-0.1994	0.1003	0.3009	0	0	0	0	0	0
3	0.17	-0.1294	0.1717	0.30369	0.3067	0	0	0	0	0
4	0.25	-0.0494	0.2553	0.30692	0.30799	0.30831	0	0	0	0
5	0.3	6 · 10 ⁻⁴	0.3093	0.30865	0.30864	0.30864	0.30864	0	0	0
6	0.36	0.0606	0.3764	0.30863	0.30864	0.30864	0.30864	0	0	0
7	0.43	0.1306	0.4218	0.3371	0.3085	0.3086	0.30862	0.30863	0.30863	0
8	0.49	0.1906	0.4536	0.35258	0.32988	0.30843	0.30857	0.3086	0.30861	0.30862

ци M :

2.10. Интерполяционный многочлен Ньютона с конечными разностями

Если интерполируемая функция задана на таблице с постоянным шагом $h (x_i = x_0 + ih, i = 0, 1, \dots, n)$, то можно использовать связь между конечными и разделенными разностями: $f(x_i; x_{i+1}; \dots; x_{i+k}) = \frac{\Delta^k y_i}{h^k k!}$. В этом случае многочлен Ньютона можно записать несколько в ином виде:

$$\text{Пусть } q = \frac{x - x_0}{h}, \quad h = x_1 - x_0 = \text{const}, \quad P_n(x) = y_0 + f(x_0; x_1)(x - x_0) + \\ + f(x_0; x_1; x_2)(x - x_0)(x - x_1) + f(x_0; x_1; x_2; x_3)(x - x_0)(x - x_1)(x - x_2) + \dots$$

$$\text{Преобразуем разделенные разности в конечные: } f(x_0; x_1) = \frac{\Delta y_0}{1! h}, \quad \text{тогда} \\ f(x_0; x_1)(x - x_0) = \frac{\Delta y_0}{1!} \cdot q, \quad f(x_0; x_1; x_2)(x - x_0)(x - x_1) = \frac{\Delta^2 y_0}{2!} \frac{(x - x_0)}{h} \frac{(x - x_1)}{h} = \\ = \frac{\Delta^2 y_0}{2!} q \frac{(x - x_0) + (x_0 - x_1)}{h} = \frac{\Delta^2 y_0}{2!} q(q-1), \quad f(x_0; x_1; x_2; x_3)(x - x_0)(x - x_1)(x - x_2) =$$

$$= \frac{\Delta^3 y_0}{3!} q(q-1) \frac{x - x_0 + x_0 - x_2}{h} = \frac{\Delta^3 y_0}{3!} q(q-1)(q-2) \text{ и так далее.}$$

Тогда многочлен Ньютона можно переписать в следующем виде:

$$P_n(x) = y_0 + \frac{\Delta y_0}{1!} q + \frac{\Delta^2 y_0}{2!} q(q-1) + \frac{\Delta^3 y_0}{3!} q(q-1)(q-2) + \dots + \frac{\Delta^n y_0}{n!} q(q-1)(q-2)\dots(q-n+1). \quad (2.10.1)$$

Эту формулу называют интерполяционным многочленом Ньютона с конечными разностями для интерполяции вперед. В ней используются только конечные разности, расположенные в верхней косой строке таблицы конечных разностей. Если использовать разности нижней косой строки, то аналогично получим многочлен Ньютона с конечными разностями для интерполяции назад:

$$\begin{aligned} P_n(x) = & y_n + \frac{\Delta y_{n-1}}{1!} q + \frac{\Delta^2 y_{n-2}}{2!} q(q+1) + \frac{\Delta^3 y_{n-3}}{3!} q(q+1)(q+2) + \dots \\ & + \frac{\Delta^n y_0}{n!} q(q+1)(q+2)\dots(q+n-1). \end{aligned} \quad (2.10.2)$$

Пример. Вычислить $\sin t = \int_0^x \frac{\sin t}{t} dt$ в точках $x^{(1)} = 0.1$ и $x^{(2)} = 0.7$, используя формулы (2.10.1) и (2.10.2) при $n = 3$, если

x	$\sin t$	Δy	$\Delta^2 y$	$\Delta^3 y$	$\Delta^4 y$	$\Delta^5 y$
0.0	0.00000					
		19956				
0.2	0.19956		-266			
		19690		-257		
0.4	0.39646		-523		10	
		19167		-247		8
0.6	0.58813		-770		18	
		18397		-229		
0.8	0.77210		-999			
		17398				
1.0	0.94608					

Здесь $x_0 = 0.0$, $h = 0.2$, $q = \frac{x - x_0}{h}$. Для $x^{(1)}$ используем формулу (2.10.1), так как значение

$$x^{(1)} = 0.1 \text{ расположено в начале таблицы, тогда } q^{(1)} = \frac{0.1 - 0.0}{0.2} = \frac{1}{2}.$$

$$\begin{aligned} \sin(0.1) \approx & \sin(0.0) + \frac{\Delta y_0}{1!} q^{(1)} + \frac{\Delta^2 y_0}{2!} q^{(1)}(q^{(1)} - 1) + \frac{\Delta^3 y_0}{3!} q^{(1)}(q^{(1)} - 1)(q^{(1)} - 2) = \\ = & 0.0 + \frac{1}{2} \cdot 0.19956 + \frac{1}{2} \cdot \frac{1}{2} \cdot \left(-\frac{1}{2}\right) \cdot (-0.00266) + \frac{1}{6} \cdot \frac{1}{2} \cdot \left(-\frac{1}{2}\right) \cdot \left(-\frac{3}{2}\right) \cdot (-0.00257) = \\ = & 0.0 + 0.09978 + 0.0003325 - 0.0003212 = 0.10270. \end{aligned}$$

Для $x^{(2)} = 0.7$ лучше использовать формулу (2.10.2), так как $x^{(2)} = 0.7$ расположено ближе к нижнему краю таблицы. Тогда $x_n = 1.0$, $h = 0.2$, $q^{(2)} = \frac{x^{(2)} - x_{n-1}}{h} = \frac{0.7 - 0.8}{0.2} = -0.5$.

$$\begin{aligned}
si(0.7) &\approx si(0.8) + \frac{\Delta y_3}{1!} q^{(2)} + \frac{\Delta^2 y_2}{2!} q^{(2)}(q^{(2)} + 1) + \frac{\Delta^3 y_1}{3!} q^{(2)}(q^{(2)} + 1)(q^{(2)} + 2) = \\
&= 0.77210 + (-0.5) \cdot 0.18397 + (-0.5) \cdot (0.5) \cdot \frac{1}{2} \cdot (-0.00770) + (-0.5) \cdot (0.5) \cdot 1.5 \cdot \frac{1}{6} \times \\
&\quad \times (-0.00247) = 0.77210 - 0.091985 + 0.0009625 + 0.0001544 = 0.68123.
\end{aligned}$$

2.11. Лабораторная работа № 3. Интерполярование и экстраполирование данных. Интерполяционный многочлен Ньютона

Интерполяционный многочлен Ньютона является другой формой записи единого интерполяционного многочлена. Применение многочлена Ньютона имеет практические преимущества по сравнению с формулой Лагранжа и в случае неравноотстоящих, и особенно в случае равноотстоящих узлов.

Интерполяционные многочлены Ньютона нужных степеней строятся по формулам (2.7.1) или (2.10.1) и (2.10.2).

Пример. Найти приближенное значение функции $y = f(x)$ при данном значении аргумента x_T с помощью интерполяционного многочлена Ньютона, если функция задана в неравноотстоящих узлах таблицы и $x_T = 0.552$:

x	0.35	0.41	0.47	0.51	0.56	0.64
y	2.73951	2.30080	1.96464	1.78776	1.59502	1.34310

Для вычисления интерполяционного многочлена по формуле (2.7.1) необходима таблица разделенных разностей, по которой можно вычислять разности до пятого порядка включительно по формулам

$$\begin{aligned}
f(x_1; x_2) &= \frac{f(x_2) - f(x_1)}{x_2 - x_1}, \quad f(x_1; x_2; x_3) = \frac{f(x_2; x_3) - f(x_1; x_2)}{x_3 - x_1}, \\
... f(x_1; x_2; x_3; x_4; x_5; x_6) &= \frac{f(x_2; x_3; x_4; x_5; x_6) - f(x_1; x_2; x_3; x_4; x_5)}{x_6 - x_1}.
\end{aligned}$$

Запишем получаемые конечные разности в виде таблицы, однако, в целях экономии места, не будем писать разности между строк x_i и x_{i+1} . Тогда получим

x_i	$f(x_i)$	$f(x_i; x_{i+1})$	$f(x_i; ... x_{i+2})$	$f(x_i; ... x_{i+3})$	$f(x_i; ... x_{i+4})$	$f(x_i; ... x_{i+5})$
0.35	2.73951	-7.311833	14.243057	-15.227431	-102.232694	825.705728
0.41	2.30080	-5.602667	11.806667	-36.696296	137.221967	
0.47	1.96464	-4.422000	6.302222	-5.135244		
0.51	1.78776	-3.854800	5.429231			
0.56	1.59502	-3.149000				
0.64	1.34310					

Далее по формуле (2.7.1) вычисляем

$$P_5(x) = \sum_{k=1}^6 f(x_1; x_2; \dots; x_k) \omega_k(x) = f(x_1) + f(x_1; x_2)(x - x_1) + \dots + f(x_1; x_2; \dots; x_6)(x - x_1) \dots (x - x_5) =$$

$$\begin{aligned}
&= 2.739510 - 7.311833(x - 0.35) + 14.243057(x - 0.35)(x - 0.41) - 15.227431(x - 0.35) \times \\
&\quad \times (x - 0.41)(x - 0.47) - 102.232694(x - 0.35)(x - 0.41)(x - 0.47)(x - 0.51) + 825.705728 \times \\
&\quad \times (x - 0.35)(x - 0.41) \cdot (x - 0.47) \cdot (x - 0.51)(x - 0.56) = 825.705728x^5 - 2001.355769x^4 + \\
&\quad + 1898.621006x^3 - 870.466826x^2 + 184.903812x - 11.051930.
\end{aligned}$$

Подставляя теперь значение $x_T = 0.552$ вместо аргумента x в найденную формулу, вычислим приближенное значение функции $P_5(0.552) = 1.62450$.

Задание № 1. Построить интерполяционный многочлен Ньютона по неравноотстоящей сетке узлов и найти приближенное значение интерполируемой функции $y = f(x)$ при значении аргумента x_T .

I.

№ варианта	x_T	x	y
1	0.154	0.05	4.4817
6	0.257	0.19	4.9530
11	0.291	0.21	5.4739
16	0.335	0.27	6.0496
21	0.367	0.32	6.6859
26	0.412	0.34	7.3891
		0.39	8.1662
		0.45	9.0250

II.

№ варианта	x_T	x	y
2	0.015	0.01	9.9182
7	0.195	0.11	9.5194
12	0.137	0.16	9.1365
17	0.245	0.23	8.8769
22	0.385	0.28	8.4164
27	0.478	0.39	8.0779
		0.46	7.7530
		0.50	7.4412

III.

№ варианта	x_T	x	y
3	1.330	1.30	4.7556
8	1.455	1.45	5.3533
13	1.813	1.65	6.4552
18	2.290	1.90	7.5618
23	2.750	2.40	8.6734
28	3.040	2.55	9.7904
		2.80	10.9131
		3.20	12.0419

IV.

№ варианта	x_T	x	y
4	3.905	3.50	33.1154
9	4.110	4.55	34.8133
14	5.995	5.60	36.5982
19	7.213	6.20	38.4747
24	9.115	7.75	40.4473
29	10.050	8.80	42.5211
		9.45	44.7012
		10.95	46.9931

V.

№ варианта	x_T	x	y
5	1.090	1.01	12.6183
10	1.250	1.08	12.7644
15	1.275	1.11	12.9122
20	1.316	1.21	13.0617
25	1.488	1.26	13.2130
30	1.343	1.33	13.3660
		1.46	13.5207
		1.51	13.8357

Конечные разности практически вычисляются значительно проще разделенных, поэтому исходная таблица может содержать заметно больше узлов при тех же вычислительных затратах.

Пример. Используя интерполяционную формулу Ньютона с конечными разностями для интерполяции вперед и назад, вычислить значения функции при данных значениях аргумента $x_{T_1} = 0.4675$ и $x_{T_2} = 0.5410$.

x	0.45	0.46	0.47	0.48	0.49	0.50
y	20.1946	19.6133	18.9425	18.1746	17.3010	16.3123
x	0.51	0.52	0.53	0.54	0.55	0.56
y	15.1984	13.9484	12.5508	10.9937	9.2647	7.3510

Составим таблицу конечных разностей. Ограничимся при этом разностями четвертого порядка, так как они практически постоянны. Для вычисления значения функции при $x_{T_1} = 0.4675$ воспользуемся формулой Ньютона для интерполяции вперед (2.10.1). При этом разности из таблицы лучше всего брать стоящие по той диагонали, которая ближе всего расположена к x_{T_1} . В таблице разностей эта диагональ подчеркнута. Тогда

$$x_0 = 0.46, q = \frac{x - x_0}{h} = \frac{0.4675 - 0.46}{0.01} = 0.75 \quad \text{и} \quad f(0.4675) \approx P(0.4675) = f(x_0) + \frac{\Delta y_0}{1!} q + \frac{\Delta^2 y_0}{2!} q(q-1) + \frac{\Delta^3 y_0}{3!} q(q-1)(q-2) + \frac{\Delta^4 y_0}{4!} q(q-1)(q-2)(q-3) = 19.6133 - \frac{0.6708}{1!} \cdot 0.75 - \frac{0.0971}{2!} \cdot 0.75 \cdot (-0.25) - \frac{0.0086}{3!} \cdot 0.75 \cdot (-0.25) \cdot (-1.25) - \frac{0.0008}{4!} \cdot 0.75 \cdot (-0.25) \cdot (-1.25) \cdot (-2.25) = 19.6133 - 0.3354 + 0.0091 - 0.0067 + 0.0000 = 19.2803$$

с точностью $\varepsilon = 10^{-4}$.

x	y	Δy	$\Delta^2 y$	$\Delta^3 y$	$\Delta^4 y$
0.45	20.1946				

		-0.5813			
0.46	19.6133		-0.0895		
		-0.6708		-0.0076	
0.47	18.9425		-0.0971		-0.0010
		-0.7679		-0.0086	
0.48	18.1746		-0.1057		-0.0008
		-0.8736		-0.0094	
0.49	17.3010		-0.1151		-0.0007
		-0.9887		-0.0101	
0.50	16.3123		-0.1252		-0.0008
		-1.1139		-0.0109	
0.51	15.1984		-0.1361		-0.0006
		-1.2500		-0.0115	
0.52	13.9484		-0.1476		-0.0004
		-1.3976		-0.0119	
0.53	12.5508		-0.1595		-0.0005
		-1.5571		-0.0124	
0.54	10.9937		-0.1719		-0.0004
		-1.7290		-0.0128	
0.55	9.2647		-0.1847		
		-1.9137			
0.56	7.3510				

Аналогично, для $x_{T_2} = 0.5410$, $x_n = 0.55$, $q = \frac{x - x_n}{h} = \frac{0.5410 - 0.55}{0.01} = -0.9$. Воспользуемся формулой (2.10.2) для интерполяции назад:

$$\begin{aligned}
 f(0.5410) \approx P(0.5410) &= f(x_n) + \frac{\Delta y_{n-1}}{1!} q + \frac{\Delta^2 y_{n-2}}{2!} q(q+1) + \frac{\Delta^3 y_{n-3}}{3!} q(q+1)(q+2) + \\
 &+ \frac{\Delta^4 y_{n-4}}{4!} q(q+1)(q+2)(q+3) = 9.2647 - \frac{1.7290}{1!} \cdot (-0.9) - \frac{0.1719}{2!} \cdot (-0.9) \cdot -\frac{0.0124}{3!} \times \\
 &\times (-0.9) \cdot 0.1 \cdot 1.1 - \frac{0.0005}{4!} \cdot (-0.9) \cdot 0.1 \cdot 1.1 \cdot 2.1 = 9.2647 + 1.5561 + 0.0077 + 0.0002 + \\
 &+ 0.0000 = 10.8287
 \end{aligned}$$

с той же точностью, что и в предыдущей формуле.

Задание № 2. Вычислить приближенное значение функции $y = f(x)$ по интерполяционной формуле Ньютона для интерполяции вперед или назад.

I.

№ варианта	x_T	x	y
1	4.413	4.35	16.3597
6	6.085	4.60	17.7334
11	5.218	4.85	18.7686
16	6.480	5.10	20.0334
21	4.390	5.35	22.2846
26	6.020	5.60	23.5973
		5.85	25.0811
		6.10	26.5278
		6.35	28.3944
		6.60	29.9902

II.

№ варианта	x_T	x	y
2	1.210	1.15	66.1659
7	4.680	1.55	63.9989
12	2.000	1.95	61.9658
17	3.870	2.35	60.0551
22	2.411	2.75	58.2558
27	3.275	3.15	56.5583
		3.55	54.6807
		3.95	52.7220
		4.35	50.5229
		4.75	48.1091

III.

№ варианта	x_T	x	y
3	0.105	0.10	1.2618
8	0.121	0.16	1.2764
13	0.625	0.22	1.2912
18	0.438	0.28	1.3062
23	0.534	0.34	1.3213
28	0.229	0.40	1.3366
		0.46	1.3521
		0.52	1.3677
		0.58	1.3836
		0.64	1.3995

IV.

№ варианта	x_T	x	y
4	1.871	1.350	4.2556
9	1.369	1.355	4.3533
14	1.382	1.360	4.4552
19	1.394	1.365	4.5618
24	1.354	1.370	4.6734
29	1.362	1.375	4.7904
		1.380	4.9131
		1.385	5.0419
		1.390	5.1778
		1.395	5.3202

V.

№ варианта	x_T	x	y
5	1.285	1.20	10.6044
10	1.211	1.21	11.3276
15	1.272	1.22	11.9671
20	1.228	1.23	12.5324
25	1.263	1.24	13.0328
30	1.237	1.25	13.4776
		1.26	13.8759
		1.27	14.2367
		1.28	14.5688
		1.29	14.8809

Приступим к решению в пакете Mathcad заданий этой лабораторной работы. Решим уже разобранную задачу первого примера. Построить матрицу разделенных разностей можно быстро и легко с помощью дискретного аргумента. Введем с клавиатуры

$$x := \begin{pmatrix} 0.35 \\ 0.41 \\ 0.47 \\ 0.51 \\ 0.56 \\ 0.64 \end{pmatrix} \quad y := \begin{pmatrix} 2.73951 \\ 2.30080 \\ 1.96464 \\ 1.78776 \\ 1.59502 \\ 1.34310 \end{pmatrix}$$

$$M^{(1)} := x \quad M^{(2)} := y \quad i := 1 \dots 5 \quad M_{i,3} := \frac{(M_{i+1,2} - M_{i,2})}{x_{i+1} - x_i}$$

$$i := 1 \dots 4 \quad M_{i,4} := \frac{(M_{i+1,3} - M_{i,3})}{x_{i+2} - x_i}$$

$$i := 1 \dots 3 \quad M_{i,5} := \frac{(M_{i+1,4} - M_{i,4})}{x_{i+3} - x_i}$$

$$i := 1 \dots 2 \quad M_{i,6} := \frac{(M_{i+1,5} - M_{i,5})}{x_{i+4} - x_i}$$

$$M_{i,7} := \frac{(M_{2,6} - M_{1,6})}{x_6 - x_1}$$

Если теперь набрать $M =$, то на экран выведется точно такая же таблица, что и полученная нами в первом примере этой лабораторной работы.

Получить аналитический вид многочлена можно с помощью меню Symbolics (Символьные выражения) – Simplify (Упростить), набирая по формуле (2.7.1) числовые значения коэффициентов из матрицы M :

$$P1(z) := 2.73951 + (-7.311833) \cdot (z - 0.35)$$

$$5.298652 - 7.311833 \cdot z$$

$$P2(z) := 2.73951 + (-7.311833) \cdot (z - 0.35) + 14.243056 \cdot (z - 0.35) \cdot (z - 0.41)$$

$$P3(z) := 2.73951 + (-7.311833) \cdot (z - 0.35) + 14.243056 \cdot (z - 0.35) \cdot (z - 0.41) \dots$$

$$+ (-15.227431) \cdot (z - 0.35) \cdot (z - 0.41) \cdot (z - 0.47)$$

$$8.369544 - 25.760930 \cdot z + 32.972795 \cdot z^2 - 15.227431 \cdot z^3$$

В Mathcad длинный оператор, не помещающийся в одну строку на странице, можно перенести в следующую строку, нажав [Ctrl][Enter]. Это проделано в предыдущем операторе. Аналогично

$$P4(z) := 2.73951 + (-7.311833) \cdot (z - 0.35) + 14.243056 \cdot (z - 0.35) \cdot (z - 0.41) \dots$$

$$+ (-15.227431) \cdot (z - 0.35) \cdot (z - 0.41) \cdot (z - 0.47) \dots$$

$$+ (-102.232684) \cdot (z - 0.35) \cdot (z - 0.41) \cdot (z - 0.47) \cdot (z - 0.51)$$

$$4.853055 + 7.239956 \cdot z - 82.345571 \cdot z^2 + 162.657283 \cdot z^3 - 102.232594 \cdot z^4$$

$$\begin{aligned}
P5(z) := & 2.73951 + (-7.311833) \cdot (z - 0.35) + 14.243056 \cdot (z - 0.35) \cdot (z - 0.41) \dots \\
& + (-15.227431) \cdot (z - 0.35) \cdot (z - 0.41) \cdot (z - 0.47) \dots \\
& + (-102.232684) \cdot (z - 0.35) \cdot (z - 0.41) \cdot (z - 0.47) \cdot (z - 0.51) \dots \\
& + 825.705728 \cdot (z - 0.35) \cdot (z - 0.41) \cdot (z - 0.47) \cdot (z - 0.51) \cdot (z - 0.56) \\
& - 11.051930 + 184.903812 \cdot z - 870.466826 \cdot z^2 + 1898.621006 \cdot z^3 \dots \\
& + (-2001.355769) \cdot z^4 + 825.705728 \cdot z^5
\end{aligned}$$

Значение функции в требуемой точке вычисляется обращением к соответствующей подпрограмме:

$$xt := 0.552 \quad xx_1 := P1(xt) \quad xx_2 := P2(xt) \quad xx_3 := P3(xt) \quad xx_4 := P4(xt) \quad xx_5 := P5(xt)$$

$$xx = \begin{pmatrix} 1.262520 \\ 1.671068 \\ 1.635251 \\ 1.625151 \\ 1.624499 \end{pmatrix}$$

Эту же задачу можно решить средствами программирования пакета Mathcad. В среде Mathcad потребность в программировании ощущается не только для расширения и совершенствования базового набора математических инструментов, но и в учебных целях. Пакет содержит одиннадцать функций экстраполяции и интерполяции, в частности линейную интерполяцию (**linterp**) и интерполяцию сплайнами (**cspline**, **pspline**, **lspline**). Полиномиальная интерполяция осуществляется функциями **polint** и **polin2**, однако они не являются встроеными и становятся таковыми лишь после подгрузки соответствующего электронного учебника.

Составим поэтому программу, вычисляющую интерполяционный многочлен Ньютона по разделенным разностям. Эту задачу удобно разбить на три части: а) составление таблицы разделенных разностей; б) вычисление коэффициентов функции $\omega_n(x)$ в формуле (2.7.1) и в) вычисление самого многочлена $P_n(x)$.

Подпрограмма, вычисляющая таблицу разделенных разностей, очень проста и понятна:

$$\text{Newt}(x, y, n) := \begin{cases} M^{(1)} \leftarrow x \\ M^{(2)} \leftarrow y \\ \text{for } j \in 1..n-1 \\ \quad \text{for } i \in 1..n-j \\ \quad M_{i,j+2} \leftarrow \frac{(M_{i+1,j+1} - M_{i,j+1})}{x_{i+j} - x_i} \\ M \end{cases}$$

Здесь x и y - вектора аргумента и функции, n - число узлов заданной сетки. После обращения к данной подпрограмме

$$MM := \text{Newt}(x, y, 6)$$

можно еще раз напечатать полученную матрицу, набрав $MM =$. Эта таблица будет опять точно такой же, как и в первом примере.

Нахождение коэффициентов функции $\omega_n(x)$, включая раскрытие скобок и приведение подобных членов, - задача значительно более трудная. Без сомнения, можно предложить несколько вариантов этой подпрограммы, которые будут длиннее или короче друг друга в зависимости от выбранного алгоритма. Один из вариантов может быть, например, таким:

```

omega(x,n) := | if n=2
                |   | a2,2 ← a2,2
                |   | return a
                |
                | for i ∈ 3..n
                |
                |   for j ∈ i..n
                |
                |     aj,i ← aj-1,i + aj-1,i-1 · xj
                |
                |   for i ∈ 2,4..n+1
                |
                |     for j ∈ 1..n
                |
                |       aj,i ← (-1) · aj,i
                |
                | a
                |
                | b

```

dimNewt(x,y,n,m) :=

$M \leftarrow \text{Newt}(x,y,n)$ $a \leftarrow \text{omega}(x,m)$ for $i \in 1..m$ for $j \in 1..i+1$ $a_{i,j} \leftarrow a_{i,j} \cdot M_{1,i+2}$ $b_1 \leftarrow a_{m,1}$ $b_{m+1} \leftarrow \sum_{i=1}^m a_{i,i+1} + M_{1,2}$ return b if $m=1$ for $i \in 1..m-1$ $k \leftarrow m-i+1$ $b_k \leftarrow \sum_{j=1}^k a_{i+j-1,j}$	\rightarrow $\dim\text{Newt}(x,y,n,m) :=$ $M \leftarrow \text{Newt}(x,y,n)$ $a \leftarrow \text{omega}(x,m)$ for $i \in 1..m$ for $j \in 1..i+1$ $a_{i,j} \leftarrow a_{i,j} \cdot M_{1,i+2}$ $b_1 \leftarrow a_{m,1}$ $b_{m+1} \leftarrow \sum_{i=1}^m a_{i,i+1} + M_{1,2}$ return b if $m=1$ for $i \in 1..m-1$ $k \leftarrow m-i+1$ $b_k \leftarrow \sum_{j=1}^k a_{i+j-1,j}$
--	--

Подпрограмма $\text{omega}(x, n)$ выдает матрицу $a_{n \times (n+1)}$, в которой по строкам записаны коэффициенты всех функций $\omega_n(x)$ от первой степени $\omega_1(x)$ до n -степени $\omega_n(x)$, причем коэффициенты расположены начиная со старшей степени аргумента. Последняя подпрограмма PolinNewt реализует формулу (2.7.1) и выдает вектор коэффициентов интерполяционного многочлена Ньютона степени m ($1 \leq m \leq n - 1$).

Можно проверить правильность предыдущих вычислений, набрав

$a1 := \text{PolinNewt}(x, y, 6, 5)$ $a1 =$

Аналогичные программы можно составить для вычисления интерполяционного многочлена Ньютона по конечным разностям для интерполяции вперед или назад. Таблица конечных разностей вычисляется следующей подпрограммой:

```

Tabl(x,y,n) := | M'^{1} <- x
                  | M'^{2} <- y
                  | for j ∈ 1..n-1
                  |   for i ∈ 1..n-j
                  |     M_{i,j+2} ← M_{i+1,j+1} - M_{i,j+1}
                  |
                  | M

```

Структура формулы (2.10.1) позволяет при вычислении произведений вида $q \cdot (q-1)(q-2)\dots(q-n+1)$ использовать уже готовую функцию $\omega_n(x)$. Действительно, $q = \frac{x-x_1}{h} = \frac{1}{h}(x-x_1) = \frac{1}{h}\omega_1(x)$, $q(q-1) = \frac{x-x_1}{h} \cdot \frac{x-x_2}{h} = \frac{1}{h^2}\omega_2(x)$ и так далее.

Подпрограмма, реализующая формулу Ньютона для интерполяции вперед, может быть, например, такой:

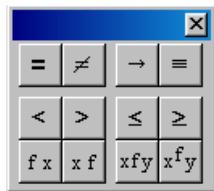
```

PolinNewtConst(x,y,n,m,t) := | M ← Tabl(x,y,n)
                                | a ← omega(x,m)
                                | h ← x_2 - x_1
                                | h1 ← h
                                | c ← 1
                                | for i ∈ 1..n-1 if m > 2
                                |   if | M_{1,i+2} | < t
                                |     | m ← i-1
                                |     | break
                                |   for i ∈ 1..m
                                |     | c ← c · i
                                |     | b_i ← M_{1,i+2} / c · h1
                                |     | h1 ← h1 · h
                                |     | for j ∈ 1..i+1
                                |       | a_{i,j} ← a_{i,j} · b_i
                                |   b_1 ← a_{m,1}
                                |
                                | b_{m+1} ← ∑_{i=1}^m a_{i,i+1} + M_{1,2}
                                |
                                | return b if m=1
                                | for i ∈ 1..m-1
                                |   | k ← m-i+1
                                |   | b_k ← ∑_{j=1}^k a_{i+j-1,j}
                                |
                                | b

```

При написании этой подпрограммы использовались операторы **break** – выхода из текущего цикла и операции отношения $>$, $<$. Операции отношения берутся из панели равенств и отношений (Evaluation and Boolean Palette), которую открывает кнопка .

Эта панель имеет вид



Щелчком по кнопкам этой панели в рабочий документ вставляются соответствующие знаки отношений \ll , \gg , \neq , \rightarrow и т.п.

В список параметров новой подпрограммы включен параметр ε - заданная точность вычислений. Это связано с тем, что конечные разности в отличие от разделенных быстро убывают, особенно для достаточно гладких функций. Если задать очень высокую степень аргумента, то в формуле (2.10.1) будет использована конечная разность соответствующей степени, которая может быть очень маленькой, меньше заданной точности ε . При малых шагах сетки h быстро накапливаются большие ошибки, которые могут совершенно исказить конечный результат. Поэтому в подпрограмме анализируется таблица конечных разностей по заданной точности ε и соответственно уменьшается степень интерполяционного многочлена.

Введем с клавиатуры

$$x := \begin{pmatrix} 0.01 \\ 0.06 \\ 0.11 \\ 0.16 \\ 0.21 \\ 0.26 \end{pmatrix} \quad y := \begin{pmatrix} 9.9182 \\ 9.5193 \\ 9.1365 \\ 8.7691 \\ 8.4164 \\ 8.0779 \end{pmatrix} \quad M2 := \text{Tabl}(x, y, 6)$$

$$M2 = \begin{pmatrix} 0.01 & 9.9182 & -0.3989 & 0.0161 & -7 \cdot 10^{-4} & 1.7763568 \cdot 10^{-15} & 2 \cdot 10^{-4} \\ 0.06 & 9.5193 & -0.3828 & 0.0154 & -7 \cdot 10^{-4} & -2 \cdot 10^{-4} & 0 \\ 0.11 & 9.1365 & -0.3674 & 0.0147 & -5 \cdot 10^{-4} & 0 & 0 \\ 0.16 & 8.7691 & -0.3527 & 0.0142 & 0 & 0 & 0 \\ 0.21 & 8.4164 & -0.3385 & 0 & 0 & 0 & 0 \\ 0.26 & 8.0779 & 0 & 0 & 0 & 0 & 0 \end{pmatrix}$$

$$\varepsilon := 10^{-6} \quad a2 := \text{Polin Newt Const}(x, y, 6, 5, \varepsilon)$$

$$a2 = \begin{pmatrix} -0.933333 \\ 3.388 \\ -8.211147 \\ 9.999974 \end{pmatrix}$$

Таким образом, восстановлен только многочлен третьей степени (по заданной точности ε), который имеет вид

$$P3(x) := -0.933333 \cdot x^3 + 3.388 \cdot x^2 - 8.211147 \cdot x + 9.999974.$$

Это многочлен для интерполяции вперед, его можно употреблять для нахождения приближенных значений функции в верхней половине таблицы конечных разностей $M2$. Например, $P3(0.05) = 9.59$, $P3(0.09) = 9.263$. Аналогичную формулу для интерполяции назад можно легко получить на основе приведенного материала. Желающие могут проделать это самостоятельно.

2.12. Интерполяционные формулы Гаусса, Стирлинга и Бесселя

Формулы Гаусса*. Выпишем подробно таблицу конечных разностей для того, чтобы отмечать задействованные элементы этих таблиц в далее рассматриваемых формулах.

x	y	Δy	$\Delta^2 y$	$\Delta^3 y$	$\Delta^4 y$	$\Delta^5 y$	$\Delta^6 y$
x_{-4}	y_{-4}						
		Δy_{-4}					
x_{-3}	y_{-3}		$\Delta^2 y_{-4}$				
		Δy_{-3}		$\Delta^3 y_{-4}$			
x_{-2}	y_{-2}		$\Delta^2 y_{-3}$		$\Delta^4 y_{-4}$		
		Δy_{-2}		$\Delta^3 y_{-3}$		$\Delta^5 y_{-4}$	
x_{-1}	y_{-1}		$\Delta^2 y_{-2}$		$\Delta^4 y_{-3}$		$\Delta^6 y_{-4}$
x_0	y_0	Δy_{-1}	$\Delta^2 y_{-1}$	$\Delta^3 y_{-2}$	$\Delta^4 y_{-3}$	$\Delta^5 y_{-4}$	$\Delta^6 y_{-3}$
x_1	y_1	Δy_0	$\Delta^2 y_0$	$\Delta^3 y_{-1}$	$\Delta^4 y_{-2}$	$\Delta^5 y_{-3}$	$\Delta^6 y_{-2}$
x_2	y_2	Δy_1	$\Delta^2 y_1$		$\Delta^4 y_0$		
		Δy_2		$\Delta^3 y_1$		$\Delta^5 y_{-1}$	
x_3	y_3		$\Delta^2 y_2$				
		Δy_3					
x_4	y_4						

- - вторая интерполяционная формула Гаусса,
- - первая интерполяционная формула Гаусса,
- → - формула Бесселя.

Первая интерполяционная формула Гаусса для интерполяции вперед имеет вид

* Карл Фридрих Гаусс (1777-1855) - немецкий математик, астроном и геодезист.

$$\begin{aligned}
P(x) = & y_0 + \frac{\Delta y_0}{1!} q + \frac{\Delta^2 y_{-1}}{2!} q(q-1) + \frac{\Delta^3 y_{-1}}{3!} (q-1)q(q+1) + \frac{\Delta^4 y_{-2}}{4!} (q-2)(q-1)q(q+1) + \\
& + \frac{\Delta^5 y_{-2}}{5!} (q-2)(q-1)q(q+1)(q+2) + \dots + \frac{\Delta^{2n-1} y_{-(n-1)}}{(2n-1)!} (q-n+1)(q-n+2)\dots(q+n-1) + \\
& + \frac{\Delta^{2n} y_{-n}}{(2n)!} (q-n)(q-n+1)\dots(q+n-1).
\end{aligned} \quad (2.12.1)$$

Формулы Гаусса применяются для интерполяции в середине таблицы вблизи точки x_0 . При этом формула (2.12.1) применяется при $x > x_0$, а формула (2.12.2) - вторая интерполяционная формула Гаусса - при $x < x_0$.

$$\begin{aligned}
P(x) = & y_0 + \frac{\Delta y_{-1}}{1!} q + \frac{\Delta^2 y_{-1}}{2!} q(q+1) + \frac{\Delta^3 y_{-2}}{3!} (q-1)q(q+1) + \frac{\Delta^4 y_{-2}}{4!} (q-1)q(q+1)(q+2) + \\
& + \dots + \frac{\Delta^{2n-1} y_{-n}}{(2n-1)!} (q-n+1)(q-n+2)\dots(q+n-1) + \frac{\Delta^{2n} y_{-n}}{(2n)!} (q-n+1)(q-n+2)\dots(q+n), \\
q = & \frac{x - x_0}{h}.
\end{aligned} \quad (2.12.2)$$

В обеих формулах используются разности, расположенные вблизи середины таблицы (они помечены). Остаточный член формул (2.12.1) и (2.12.2) может быть записан в виде

$$R_{2n} = \frac{h^{2n+1} f^{(2n+1)}(\zeta)}{(2n+1)!} q(q^2 - 1^2)(q^2 - 2^2)\dots(q^2 - n^2), \quad \zeta \in (a, b), \quad x_0, x_{\pm 1}, \dots, x_{\pm n} \in (a, b). \quad (2.12.3)$$

Интерполяционная формула Стирлинга* представляет собой среднее арифметическое (2.12.1) и (2.12.2), то есть среднее арифметическое первой и второй формулы Гаусса:

$$\begin{aligned}
P(x) = & y_0 + \frac{\Delta y_0 + \Delta y_{-1}}{2} q + \frac{\Delta^2 y_{-1}}{2!} q^2 + \frac{\Delta^3 y_{-2} + \Delta^3 y_{-1}}{2 \cdot 3!} (q^2 - 1^2)q + \frac{\Delta^4 y_{-2}}{4!} (q^2 - 1^2)q^2 + \\
& + \frac{\Delta^5 y_{-3} + \Delta^5 y_{-2}}{2 \cdot 5!} (q^2 - 2^2)q(q^2 - 1^2) + \dots + \frac{\Delta^{2n-1} y_{-n} + \Delta^{2n-1} y_{-(n-1)}}{2 \cdot (2n-1)!} [q^2 - (n-1)^2] [q^2 - n^2] \dots q^2 + \\
& + \frac{\Delta^{2n} y_{-n}}{(2n)!} [q^2 - (n-1)^2] \dots [q^2 - 1^2] q^2.
\end{aligned} \quad (2.12.4)$$

Остаточный член этой формулы имеет тот же вид, что и для формул Гаусса, то есть вид (2.12.3). Формула (2.12.4) применяется для интерполяции в середине таблицы при значениях q , близких к нулю. Практически ее используют при $|q| \leq 0.25$.

Интерполяционная формула Бесселя** имеет вид:

$$\begin{aligned}
P(x) = & \frac{y_0 + y_1}{2} + \Delta y_0 \left(q - \frac{1}{2} \right) + \frac{\Delta^2 y_{-1} + \Delta^2 y_0}{2 \cdot 2!} q(q-1) + \frac{\Delta^3 y_{-1}}{3!} (q-1) \left(q - \frac{1}{2} \right) q + \\
& + \frac{\Delta^4 y_{-2} + \Delta^4 y_{-1}}{2 \cdot 4!} (q-2)(q-1)q(q+1) + \dots + \frac{\Delta^{2n} y_{-n} + \Delta^{2n} y_{-n+1}}{2 \cdot (2n)!} q(q-1)(q+1)\dots(q+n-1) + \\
& + \frac{\Delta^{2n+1} y_{-n}}{(2n+1)!} \left(q - \frac{1}{2} \right) q(q-1)(q+1)(q-2)(q+2)\dots(q-n)(q+n-1).
\end{aligned} \quad (2.12.5)$$

Ее остаточный член можно записать в виде

$$R_{2n} = \frac{h^{2n+2} f^{(2n+2)}(\zeta)}{(2n+2)!} q(q^2 - 1^2)(q^2 - 2^2)\dots(q^2 - n^2)(q-n-1), \quad \zeta \in (a, b), \quad x_0, x_{\pm 1}, \dots, x_{\pm n} \in (a, b). \quad (2.12.6)$$

Формула Бесселя используется для интерполяции в середине таблицы при значениях q , близких к 0.5, практически при $0.25 \leq q \leq 0.75$. Наиболее простой вид имеет формула (2.12.5) при $q = 0.5$, так как все члены, содержащие разности нечетного порядка, про-

* Джемс Стирлинг (1692-1770) - английский математик.

** Фридрих Вильгельм Бессель (1784-1846) - немецкий астроном и геодезист.

падают. Этот специальный случай формулы Бесселя называют формулой интерполяирования на середину. Ее обычно используют для уплотнения таблиц. Общая особенность формул (2.12.1), (2.12.2), (2.12.4) и (2.12.5) заключается в том, что слагаемые в них убывают значительно быстрее, чем в формулах Ньютона, поэтому для достижения заданной точности нужно меньше вычислений.

Пример. Используя формулу Бесселя, уплотнить таблицу значений функции $y = f(x)$ в два раза.

i	x	$f(x)$	Δy	$\Delta^2 y$	$\Delta^3 y$	$\Delta^4 y$
-2	2.0	1.5906				
			1549			
-1	2.1	1.7455		137		
			1686		14	
0	2.2	1.9141		151		
			1837		15	
1	2.3	2.0978		166		
			2003		17	
2	2.4	2.2981		183		
			2186		4	
3	2.5	2.5167				

Выпишем формулу (2.12.5) при $q = \frac{1}{2}$. Получим $P(x) = \frac{y_0 + y_1}{2} - \frac{\Delta^2 y_{-1} + \Delta^2 y_0}{16} + \frac{3(\Delta^4 y_{-2} + \Delta^4 y_{-1})}{256}$. По имеющейся таблице и по только что написанной формуле можно получить значение функции только для $x = 2.25$. Тогда $q = \frac{2.25 - 2.20}{0.1} = 0.5$ и $f(2.25) = 2.0060 - 0.0020 + 0.0000 = 2.0040$. По данной таблице можно получить еще два значения функции для $x = 2.15$ и $x = 2.35$, так как вклад последнего члена формулы для $P(x)$ меньше $4 \cdot 10^{-6}$, поэтому для $x = 2.15$ и $x = 2.35$ четвертые разности не понадобятся (их нет для значений $x = 2.15$ и $x = 2.35$). Структура формулы для $P(x)$ для этих значений не изменится, только разности, входящие в формулу, сдвинутся по таблице для $x = 2.15$ вверх, а для $x = 2.35$ - вниз на одну позицию (для $x = 2.15$ они обведены пунктиром):

$$f(2.15) = \frac{1.9141 + 1.7455}{2} - \frac{0.0137 + 0.0151}{16} = 1.8298 - 0.0018 = 1.8280 \text{ и}$$

$$f(2.35) = \frac{2.2981 + 2.0978}{2} - \frac{0.0166 + 0.0183}{16} = 2.1980 - 0.0022 = 2.1958.$$

3. МЕТОД НАИМЕНЬШИХ КВАДРАТОВ И СПЕЦИАЛЬНЫЕ ИНТЕРПОЛЯЦИОННЫЕ МНОГОЧЛЕНЫ

3.1. Постановка задачи и вывод формул метода наименьших квадратов

Задача наименьших квадратов возникает в самых различных областях науки и техники, например, к ней приходят при статистической обработке экспериментальных данных. Пусть функция $y = f(x)$ задана таблицей приближенных значений $y_i \approx f(x_i)$, $i = \overline{0, n}$, полученных с ошибками $\varepsilon_i = y_i^\circ - y_i$, где $y_i^\circ = f(x_i)$. Предположим, что для аппроксимации функции $y = f(x)$ используется линейная модель: $y = \Phi_m(x) = a_0\varphi_0(x) + a_1\varphi_1(x) + \dots + a_m\varphi_m(x)$, где $\varphi_0(x), \varphi_1(x), \dots, \varphi_m(x)$ - заданные базисные функции, a_0, a_1, \dots, a_m - параметры модели, являющиеся одновременно коэффициентами обобщенного многочлена. Часто используется одна из наиболее простых моделей $y = P_m(x) = a_0 + a_1x + \dots + a_mx^m$ - полиномиальная модель.

В случае, когда уровень неопределенности исходных данных высок, нет смысла требовать точного совпадения значений обобщенного многочлена $\Phi_m(x)$ в точках x_i с заданными значениями y_i , то есть использовать интерполяцию. Кроме того, при интерполяции происходит повторение ошибок наблюдений, в то время как при обработке экспериментальных данных желательно сглаживание ошибок. Тем не менее нужно потребовать, чтобы

$$\begin{cases} a_0\varphi_0(x_0) + a_1\varphi_1(x_0) + \dots + a_m\varphi_m(x_0) \approx y_0, \\ a_0\varphi_0(x_1) + a_1\varphi_1(x_1) + \dots + a_m\varphi_m(x_1) \approx y_1, \\ \dots \\ a_0\varphi_0(x_n) + a_1\varphi_1(x_n) + \dots + a_m\varphi_m(x_n) \approx y_n. \end{cases} \quad (3.1.1)$$

Эта же система в матричной форме имеет вид $\bar{P}\bar{a} \approx \bar{y}$. (3.1.2)

Существуют разные дополнительные критерии, позволяющие решить эту систему, так как в общем случае при $m < n$ она, вообще говоря, несовместна. Выбор \bar{a} , позволяющий наилучшим образом удовлетворить (3.1.2) в методе наименьших квадратов, состоит в том минимизируется среднее квадратическое уклонение

$$\delta(\Phi_m, y) = \sqrt{\frac{1}{n+1} \sum_{i=0}^n (\Phi_m(x_i) - y_i)^2}. \quad (3.1.3)$$

Итак, линейная задача метода наименьших квадратов состоит в следующем. Надо найти обобщенный многочлен $\Phi_m(x)$, для которого среднеквадратическое уклонение $\delta(\Phi_m, y) \Rightarrow \min$. Этот многочлен называется *многочленом наилучшего среднего квадратического приближения*. Так как набор функций $\{\varphi_i(x)\}_{i=0}^m$ всегда заранее определен, задача заключается в нахождении вектора $\bar{a} = (a_0^*, a_1^*, \dots, a_m^*)^T$ при условии $\delta(\Phi_m, y) \Rightarrow \min$. Для решения нашей задачи воспользуемся общим приемом дифференциального исчисления, а именно выпишем необходимые условия экстремума функции нескольких переменных (приравняем частные производные нулю):

$$\begin{cases} \frac{\partial S}{\partial a_k} = 0, \quad k = 0, 1, \dots, m, \text{ где} \\ S = \sum_{i=0}^n \sum_{j=0}^m (a_j \varphi_j(x_i) - y_i)^2. \end{cases} \quad (3.1.4)$$

Тогда получим $\frac{\partial S}{\partial a_k} = 2 \sum_{i=0}^n \sum_{j=0}^m (a_j \varphi_j(x_i) - y_i) \cdot \varphi_k(x_i) - 0, \quad k = \overline{0, m}$. Изменим в первом слагаемом порядок суммирования:

$$\sum_{j=0}^m \left(\sum_{i=0}^n \varphi_j(x_i) \cdot \varphi_k(x_i) \right) a_j = \sum_{i=0}^n y_i \varphi_k(x_i), \quad k = \overline{0, m}. \quad (3.1.5)$$

Уравнение (3.1.5) называется нормальной системой метода наименьших квадратов.

Если вернуться к обозначениям формулы (3.1.2), то, как нетрудно видеть, систему (3.1.5) можно записать в виде

$$P_{m \times n}^T P_{n \times m} = P_{m \times m}^T \bar{y}. \quad (3.1.6)$$

Матрица $P_{m \times n}^T P_{n \times m} = \Gamma_{m \times m}$ называется матрицей Грама*. Если еще ввести вектор $\bar{b} = P^T \bar{y}$, то система (3.1.6) перепишется в виде $\bar{\Gamma} \bar{a} = \bar{b}$ - система линейных уравнений относительно вектора \bar{a} . Можно показать, что если среди точек x_0, x_1, \dots, x_n нет совпадающих и $m \leq n$, то определитель системы (3.1.6) отличен от нуля, и, следовательно, эта система имеет единственное решение: $a_0 = a_0^*, a_1 = a_1^*, \dots, a_m = a_m^*$. Обобщенный полином с такими коэффициентами будет обладать минимальным средним квадратическим отклонением $\delta(\Phi_m, y)$.

Если $m = n$, то обобщенный многочлен, если система функций $\{\varphi_i\}_{i=0, m}$ степенная, совпадает с полиномом Лагранжа для системы точек x_0, x_1, \dots, x_n , причем $S_{\min} = 0$. При $m < n$ построение такого точного интерполяционного многочлена невозможно. Таким образом, аппроксимация функций представляет собой более общий процесс, чем интерполирование.

Если $\{\varphi_i(x)\}_{i=0}^m = \{x^i\}_{i=0}^m$, то нормальная система (3.1.5) принимает следующий вид:

$$\begin{cases} S = \sum_{i=0}^n \sum_{j=0}^m (a_j x_i^j - y_i)^2, \\ \frac{\partial S}{\partial a_k} = 2 \sum_{i=0}^n \sum_{j=0}^m (a_j x_i^j - y_i) x_i^k = 0, \\ \sum_{j=0}^m \left(\sum_{i=0}^n x_i^{j+k} \right) a_j = \sum_{i=0}^n y_i x_i^k, \quad k = \overline{0, m}. \end{cases} \quad (3.1.7)$$

Запишем систему (3.1.7) в развернутом виде в двух наиболее простых случаях при

* Иорген Педерсен Грам (1850-1916) - датский математик.

$m = 1$ и $m = 2$. В случае, когда приближение осуществляется многочленом первой степени $P_1(x) = a_0 + a_1x$, уравнения метода наименьших квадратов имеют следующий вид:

$$S = \sum_{i=0}^n \sum_{j=0}^1 (a_j x_i^j - y_i)^2 = \sum_{i=0}^n (a_0 + a_1 x_i - y_i)^2 \Rightarrow \min,$$

$$\frac{\partial S}{\partial a_k} = 2 \sum_{i=0}^n (a_0 + a_1 x_i - y_i) \cdot x_i^k = 0, \quad k = 0, 1 \text{ или}$$

$$\begin{cases} \frac{\partial S}{\partial a_0} = 2 \sum_{i=0}^n (a_0 + a_1 x_i - y_i) \cdot 1 = 0, \\ \frac{\partial S}{\partial a_1} = 2 \sum_{i=0}^n (a_0 + a_1 x_i - y_i) \cdot x_i = 0. \end{cases}$$

$$\begin{cases} a_0(n+1) + a_1 \sum_{i=0}^n x_i = \sum_{i=0}^n y_i, \\ a_0 \sum_{i=0}^n x_i + a_1 \sum_{i=0}^n x_i^2 = \sum_{i=0}^n x_i y_i \end{cases} \quad (3.1.8)$$

- нормальная система для $m = 1$ в развернутом виде. Пусть теперь $m = 2$. Аналогично получим $P_2(x) = a_0 + a_1x + a_2x^2$.

$$S = \sum_{i=0}^n \sum_{j=0}^2 (a_j x_i^j - y_i)^2 = \sum_{i=0}^n (a_0 + a_1 x_i + a_2 x_i^2 - y_i)^2 \Rightarrow \min.$$

$$\begin{cases} \frac{\partial S}{\partial a_0} = 2 \sum_{i=0}^n (a_0 + a_1 x_i + a_2 x_i^2 - y_i) \cdot 1 = 0, \\ \frac{\partial S}{\partial a_1} = 2 \sum_{i=0}^n (a_0 + a_1 x_i + a_2 x_i^2 - y_i) \cdot x_i = 0, \\ \frac{\partial S}{\partial a_2} = 2 \sum_{i=0}^n (a_0 + a_1 x_i + a_2 x_i^2 - y_i) \cdot x_i^2 = 0. \end{cases}$$

$$\begin{cases} a_0(n+1) + a_1 \sum_{i=0}^n x_i + a_2 \sum_{i=0}^n x_i^2 = \sum_{i=0}^n y_i, \\ a_0 \sum_{i=0}^n x_i + a_1 \sum_{i=0}^n x_i^2 + a_2 \sum_{i=0}^n x_i^3 = \sum_{i=0}^n x_i y_i, \\ a_0 \sum_{i=0}^n x_i^2 + a_1 \sum_{i=0}^n x_i^3 + a_2 \sum_{i=0}^n x_i^4 = \sum_{i=0}^n x_i^2 y_i \end{cases} \quad (3.1.9)$$

- нормальная система для $m = 2$ в развернутом виде для квадратичного сглаживания.

Метод вычисления параметров $a_0^*, a_1^*, \dots, a_m^*$ с помощью решения нормальной системы кажется весьма привлекательным. Действительно, задача сводится к стандартной системе линейных алгебраических уравнений с квадратной матрицей. Однако вычислительная практика показывает, что без специального выбора базисных функций $\{\varphi_i(x)\}_0^m$ уже при $m \geq 5$ нормальная система обычно оказывается плохо обусловленной. Причина в том, что система базисных функций, будучи формально независимой, на практике часто близка к линейно зависимой. Особенно этим «грешит» система степенных функций $1, x, x^2, \dots, x^m$, широко применяемая при аппроксимации алгебраическими многочленами. Лучший результат получается, если использовать систему ортогональных на отрезке $[a, b]$ функций. Пример такой системы на $[-1, 1]$ дает система многочленов Чебышева $T_0(x), T_1(x), \dots, T_m(x)$.

В настоящее время в вычислительной практике нормальная система, как правило, не используется. Применяются другие, более надежные методы, например метод сингулярного разложения матрицы P .

Пример. Пусть функция $y = f(x)$ задана следующей таблицей:

x	0.78	1.56	2.34	3.12	3.81
y	2.50	1.20	1.12	2.25	4.28

Используя метод наименьших квадратов, аппроксимируем ее многочленами первой и второй степени и найдем соответствующие средние квадратические уклонения δ_1 и δ_2 .

Вычисления, которые нужно провести, расположим по схеме, приведенной в такой таблице:

x^0	x^1	x^2	x^3	x^4	y	xy	x^2y
1	0.78	0.608	0.475	0.370	2.50	1.950	1.521
1	1.56	2.434	3.796	5.922	1.20	1.872	2.920
1	2.34	5.476	12.813	29.982	1.12	2.621	6.133
1	3.12	9.734	30.371	94.759	2.25	7.020	21.902
1	3.81	14.516	55.306	210.717	4.28	16.307	62.129
$\Sigma 5$	11.61	32.768	102.761	341.750	11.35	29.770	94.605

а) Линейная модель

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

$$\begin{cases} 5a_0 + 11.61a_1 = 11.35, \\ 11.61a_0 + 32.768a_1 = 29.770, \end{cases} \Rightarrow \begin{cases} a_0 + 2.322a_1 = 2.27, \\ 5.810a_1 = 3.415, \\ a_1 = 0.588, \\ a_0 = 0.905. \end{cases}$$

Таким образом, линейная модель имеет вид $y = 0.905 + 0.588x$.

б) Квадратичная модель

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

$$\begin{cases} 5a_0 + 11.61a_1 + 32.768a_2 = 11.35, \\ 11.61a_0 + 32.768a_1 + 102.761a_2 = 29.770, \\ 32.768a_0 + 102.761a_1 + 341.750a_2 = 94.605. \end{cases} \Rightarrow \begin{cases} a_0 + 2.322a_1 + 6.564a_2 = 2.27, \\ 5.810a_1 + 26.553a_2 = 3.415, \\ 26.674a_1 + 126.661a_2 = 20.222. \end{cases}$$

$$\begin{cases} a_0 + 2.322a_1 + 6.564a_2 = 2.27, \\ a_1 + 4.570a_2 = 0.588, \\ 4.761a_2 = 4.538. \end{cases} \Rightarrow \begin{cases} a_2 = 0.953, \\ a_1 = -3.767, \\ a_0 = 4.762. \end{cases}$$

Отсюда $y = 4.762 - 3.767x + 0.953x^2$ - вид квадратичной модели. Обе модели значительно отличаются друг от друга. Сравним исходные данные для $y = f(x)$ с соответствующими значениями y^* , полученными из обеих моделей, и вычислим δ_1 и δ_2 .

x	y	y_1^*	$y_1^* - y$	$(y_1^* - y)^2$	y_2^*	$y_2^* - y$	$(y_2^* - y)^2$
0.78	2.50	1.364	-1.136	1.290	2.404	-0.096	0.009
1.56	1.20	1.822	0.622	0.387	1.204	0.004	0.000
2.34	1.12	2.281	1.161	1.350	1.165	0.045	0.002
3.12	2.25	2.740	0.490	0.240	2.286	0.036	0.001
3.81	4.28	3.145	1.135	1.290	4.244	-0.036	0.001
			Σ	4.557		Σ	0.013

Таким образом, $\delta_1 = \sqrt{\frac{1}{5} \cdot 4.557} = 0.955$, $\delta_2 = \sqrt{\frac{1}{5} \cdot 0.013} = 0.051$. Следовательно, данным для $y = f(x)$ в исходной таблице очень хорошо соответствует квадратичная модель. Линейная модель не адекватна исходным данным и должна быть отвергнута.

3.2. Лабораторная работа № 4. Аппроксимация функции по методу наименьших квадратов

Очень часто при анализе эмпирических данных необходимо найти явную функциональную зависимость между двумя величинами x и y , полученными в результате измерений. Поскольку опытные данные всегда содержат ошибки, то строить интерполяционный многочлен $y = P_n(x)$ неrationально, так как при интерполяции ошибки повторяются. Желательно по возможности сгладить и минимизировать ошибки наблюдений. Этот результат достигается построением многочлена наилучшего среднего квадратического приближения по методу наименьших квадратов.

Итак, если $y = f(x)$ аппроксимируется многочленом вида $P_m(x) = \sum_{j=0}^m a_j x^j$, так что

система базисных функций имеет вид $\{\phi_i(x)\}_0^m = \{x^i\}_0^m$, то неизвестные коэффициенты многочлена $P_m(x)$ по методу наименьших квадратов определяются из решения системы (3.1.7).

В подразделе 3.1 описан пример «ручного» вычисления коэффициентов линейной и квадратичной модели по методу наименьших квадратов. Решим аналогичную задачу средствами пакета Mathcad различными способами. Сформируем вначале вектора исходных данных.

В алгебре матриц в среде Mathcad доступны несколько очень удобных встроенных функций, например, submatrix, stack и augment. Функция submatrix(A, m, n, k, l) извлекает из матрицы A подматрицу, содержащуюся в A со строками m по строку n и со столбца с номером k по номер l . Функции stack и augment, наоборот, формируют одну матрицу из двух. После работы stack(A, B) получается массив, сформированный расположением A над B , при этом матрицы A и B должны иметь одинаковое число столбцов. Функция augment(A, B) располагает матрицы A и B рядом, B справа от A ; эти матрицы должны иметь одинаковое число строк.

Введем с клавиатуры

$$\text{ORIGIN} := 1 \quad n := 20 \quad i := 1..n \quad x_i := \frac{i}{10}$$

$$\begin{aligned}
y1 &:= \begin{pmatrix} -3.97 & -5.83 \\ -4.07 & -6.06 \\ -4.04 & -6.40 \\ -4.30 & -6.83 \\ -4.27 & -7.54 \\ -4.54 & -7.68 \\ -4.79 & -8.36 \\ -5.07 & -8.91 \\ -5.30 & -9.34 \\ -5.51 & -9.98 \end{pmatrix} & y2 &:= \text{submatrix}(y1, 1, 10, 2, 2) & y1 &= \begin{pmatrix} -3.97 \\ -4.07 \\ -4.04 \\ -4.30 \\ -4.27 \\ -4.54 \\ -4.79 \\ -5.07 \\ -5.30 \\ -5.51 \end{pmatrix} & y2 &= \begin{pmatrix} -5.83 \\ -6.06 \\ -6.40 \\ -6.83 \\ -7.54 \\ -7.68 \\ -8.36 \\ -8.91 \\ -9.34 \\ -9.98 \end{pmatrix} \\
y &:= \text{stack}(y1, y2) & y3 &:= \text{augment}(y1, y2) & y4 &:= \text{augment}(y2, y1) & & \\
y &= \begin{pmatrix} -3.97 \\ -4.07 \\ -4.04 \\ -4.30 \\ -4.27 \\ -4.54 \\ -4.79 \\ -5.07 \\ -5.30 \\ -5.51 \\ -5.83 \\ -6.06 \\ -6.40 \\ -6.83 \\ -7.54 \\ -7.68 \\ -8.36 \\ -8.91 \\ -9.34 \\ -9.98 \end{pmatrix} & y3 &= \begin{pmatrix} -3.97 & -5.83 \\ -4.07 & -6.06 \\ -4.04 & -6.40 \\ -4.30 & -6.83 \\ -4.27 & -7.54 \\ -4.54 & -7.68 \\ -4.79 & -8.36 \\ -5.07 & -8.91 \\ -5.30 & -9.34 \\ -5.51 & -9.98 \end{pmatrix} & y4 &= \begin{pmatrix} -5.83 & -3.97 \\ -6.06 & -4.07 \\ -6.40 & -4.04 \\ -6.83 & -4.30 \\ -7.54 & -4.27 \\ -7.68 & -4.54 \\ -8.36 & -4.79 \\ -8.91 & -5.07 \\ -9.34 & -5.30 \\ -9.98 & -5.51 \end{pmatrix}
\end{aligned}$$

$$m := \text{cols}(y3) \quad m = 2 \quad n1 := \text{rows}(y4) \quad n1 = 10$$

$$\min(y) = -9.98 \quad \max(y) = -3.97 \quad \text{length}(y1) = 10 \quad \text{last}(y) = 20$$

Функции $\text{cols}(A)$ и $\text{rows}(A)$ возвращают число столбцов и строк матрицы A , $\min(A)$ и $\max(A)$ соответственно наименьшее и наибольшее значение элементов в A , $\text{length}(a)$ - число элементов в векторе \bar{a} , $\text{last}(a)$ - индекс последнего элемента в векторе \bar{a} с учетом значения переменной ORIGIN.

Построим линейную и квадратичную модель по формулам (3.1.8) и (3.1.9). Для этого вычислим следующие величины. Введем еще одну предопределенную переменную пакета Mathcad $TOL := 10^{-6}$. Она определяет допустимую погрешность для различных алгоритмов

аппроксимации, интегрирования, решения уравнений и так далее. По умолчанию $TOL = 10^{-3}$. Вычислим следующие величины:

$$x1 := \sum_{i=1}^n x_i \quad x1 = 21 \quad x2 := \sum_{i=1}^n (x_i)^2 \quad x2 = 28.7 \quad x3 := \sum_{i=1}^n (x_i)^3 \quad x3 = 44.1$$

$$x4 := \sum_{i=1}^n (x_i)^4 \quad x4 = 72.2666 \quad xy := \sum_{i=1}^n x_i \cdot y_i \quad xy = -149.812$$

$$x2y := \sum_{i=1}^n (x_i)^2 \cdot y_i \quad x2y = -222.6264 \quad y11 := \sum_{i=1}^n y_i \quad y11 = -122.79$$

$$A1 := \begin{pmatrix} 20 & 21 \\ 21 & 28.7 \end{pmatrix} \quad A1^{-1} = \begin{pmatrix} 0.215789 & -0.157895 \\ -0.157895 & 0.150376 \end{pmatrix} \quad b1 := \begin{pmatrix} -122.79 \\ -149.812 \end{pmatrix}$$

$$a1 := A1^{-1} \cdot b1 \quad a1 = \begin{pmatrix} -2.842263 \\ -3.140226 \end{pmatrix}$$

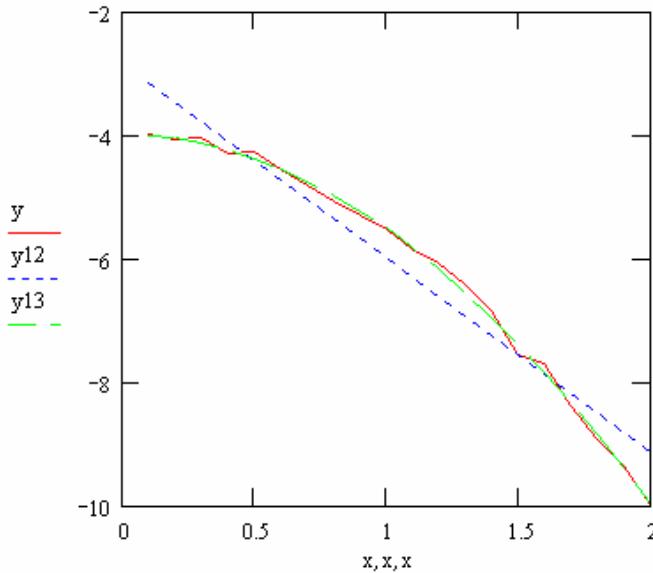
$$A := \begin{pmatrix} 20 & 21 & 28.7 \\ 21 & 28.7 & 44.1 \\ 28.7 & 44.1 & 72.2666 \end{pmatrix} \quad A^{-1} = \begin{pmatrix} 0.443509 & -1.078947 & 0.438596 \\ -1.078947 & 2.662338 & -1.196172 \\ 0.438596 & -1.196172 & 0.569606 \end{pmatrix}$$

$$b := \begin{pmatrix} -122.79 \\ -149.812 \\ -222.6264 \end{pmatrix} \quad a2 := A^{-1} \cdot b \quad a2 = \begin{pmatrix} -3.969237 \\ -0.066661 \\ -1.463602 \end{pmatrix}$$

$$i := 1 \dots n \quad y12_i := a1_1 + a1_2 \cdot x_i \quad y13_i := a2_1 + a2_2 \cdot x_i + a2_3 \cdot (x_i)^2$$

$$r1_i := (y_i - y12_i)^2 \quad r2_i := (y_i - y13_i)^2$$

$$\delta 1 := \sqrt{\frac{1}{n} \cdot \sum_{i=1}^n r1_i} \quad \delta 2 := \sqrt{\frac{1}{n} \cdot \sum_{i=1}^n r2_i} \quad \delta 1 = 0.442 \quad \delta 2 = 0.085$$



Коэффициенты системы нормальных уравнений линейной модели, то есть системы (3.1.8), находятся в матрице $A1$, коэффициенты квадратичной модели (3.1.9) - в матрице A . Решение обеих систем линейных уравнений произведено с помощью обратной матрицы. Вектор $\bar{a1}$ содержит коэффициенты линейной модели, вектор $\bar{a2}$ - квадратичной.

Далее вычисляются невязки по обеим моделям и находятся средние квадратические ошибки $\delta 1$ и $\delta 2$. Видно, что исходным данным хорошо удовлетворяет квадратичная модель $y_i = -3.969237 - 0.066661x_i - 1.463602x_i^2$. Этот факт отчетливо виден и на приведенном графике.

Mathcad не назывался бы математическим пакетом, если бы не умел решать алгебраические системы различными, в том числе и более эффективными способами. Одним из таких способов является конструкция **Given – Find**. Это две команды: **Given** (Дано) и **Find** (Найти). Сначала задается какое-нибудь начальное приближение, например, для квадратичной модели

$$\begin{pmatrix} x5 \\ y5 \\ z \end{pmatrix} := \begin{pmatrix} 1 \\ 1 \\ 1 \end{pmatrix},$$

затем за ключевым словом **Given** нужно записать анализируемую систему, связывая левые и правые части уравнений знаком «эквивалентно» (жирным знаком «равно» из панели равенств и отношений или же нажимая сразу обе клавиши [Ctrl][=]), после этого должно идти второе ключевое слово **Find**. Эта функция возвращает решение анализируемой системы:

$$\text{Given } 20 \cdot x5 + 21 \cdot y5 + 28.7 \cdot z = -122.79$$

$$21 \cdot x5 + 28.7 \cdot y5 + 44.1 \cdot z = -149.812$$

$$28.7 \cdot x5 + 44.1 \cdot y5 + 72.26666 \cdot z = -222.6264$$

$$\text{Find}(x5, y5, z) = \begin{pmatrix} -3.969237 \\ -0.066661 \\ -1.463602 \end{pmatrix}$$

Неудобство применения пары **Given – Find** в том, что решаемая система уравнений должна быть записана в скалярной форме. Вместо функции **Find** можно использовать пару **Given – MinErr**. Функция **MinErr**(x, y, \dots) дает решение системы уравнений, которое приводит к минимальным невязкам. Число неизвестных системы должно быть равно числу аргументов функции **MinErr**. В нашем случае

$$\begin{pmatrix} xx \\ yy \\ zz \end{pmatrix} := \begin{pmatrix} 1 \\ 1 \\ 1 \end{pmatrix} \quad \text{Given } 20 \cdot xx + 21 \cdot yy + 28.7 \cdot zz = -122.79$$

$$21 \cdot xx + 28.7 \cdot yy + 44.1 \cdot zz = -149.812$$

$$28.7 \cdot xx + 44.1 \cdot yy + 72.26666 \cdot zz = -222.6264$$

$$\text{MinErr}(xx, yy, zz) = \begin{pmatrix} -3.969237 \\ -0.066661 \\ -1.463602 \end{pmatrix}.$$

Наконец, для решения линейных систем алгебраических уравнений можно использовать встроенную функцию **Isolve**. Она возвращает вектор решения системы, записанной в матричном виде:

$$a22 := \text{Isolve}(A, b) \quad a22 = \begin{pmatrix} -3.969237 \\ -0.066661 \\ -1.463602 \end{pmatrix}$$

Заметим, что функцию **Isolve** можно использовать в программируемых конструкциях, тогда как пары **Given – Find** и **Given – MinErr** этого не допускают.

Приведем в заключение подпрограмму, реализующую вычисления по формуле (3.1.7) в общем случае коэффициентов сглаживающего многочлена заданной степени. Все операторы этой подпрограммы легко отождествляются с той или иной частью формулы (3.1.7). Параметры подпрограммы: x, y - вектора исходных данных, n - число точек сетки таблично заданной функции, m - требуемая степень сглаживающего многочлена. В результате работы подпрограмма МНК выдает вектор коэффициентов многочлена $P_m(x) = a_0 + a_1 x + a_2 x^2 + \dots + a_m x^m$, записанных в следующем порядке: $a_0, a_1, a_2, \dots, a_{m-1}, a_m, \delta$.

Последняя $m + 2$ компонента вектора результата содержит среднюю квадратическую ошибку представления исходных табличных данных построенным сглаживающим многочленом:

```

MHK(x,y,n,m) := | for k ∈ 1..m + 1
                   |   for j ∈ 1..m + 1
                   |     Aj,k ← ∑i=1n (xi)j+k-2
                   |     bk ← ∑i=1n yi (xi)k-1
                   |
                   | a ← lsolve(A,b)
                   | for i ∈ 1..n
                   |   y1i ← 0
                   |   for j ∈ 1..m + 1
                   |     y1i ← y1i + aj (xi)j-1
                   |
                   | r ← ∑i=1n (yi - y1i)2
                   |
                   | am+2 ← √(r/n)
                   |
                   | a

```

Для нашего примера

$$\begin{aligned}
 a23 &:= MHK(x,y,20,1) \quad a23 = \begin{pmatrix} -2.842263 \\ -3.140226 \\ 0.441879 \end{pmatrix} & a24 &:= MHK(x,y,20,2) \quad a24 = \begin{pmatrix} -3.969237 \\ -0.066661 \\ -1.463603 \\ 0.084973 \end{pmatrix} \\
 a25 &:= MHK(x,y,20,3) \quad a25 = \begin{pmatrix} -3.904312 \\ -0.397947 \\ -1.078670 \\ -0.122201 \\ 0.083012 \end{pmatrix} & a26 &:= MHK(x,y,20,4) \quad a26 = \begin{pmatrix} -3.909030 \\ -0.361037 \\ -1.153360 \\ -0.067807 \\ -0.012951 \\ 0.083006 \end{pmatrix}
 \end{aligned}$$

Видно, что для исходной таблично заданной функции многочленом наилучшего приближения является уже полученный ранее многочлен второй степени $P_2(x) = -3.969237 - 0.066661x - 1.463602x^2$. Дальнейшее усложнение модели (повышение степени многочлена) практически не изменяет среднюю квадратическую ошибку и, следовательно, не является оправданным.

Задание № 1. По методу наименьших квадратов аппроксимировать таблично заданную функцию $y = f(x)$ многочленом наилучшего среднеквадратического приближения ($1 \leq m \leq 4$).

Таблица 1

x_i	$y_i = f(x_i)$									
	Номера вариантов									
	1	2	3	4	5	6	7	8	9	10
0.1	3.15	20.95	2.20	6.00	3.13	0.07	12.28	-9.10	9.84	3.09
0.2	3.04	20.51	2.18	7.04	3.19	0.17	12.53	-9.21	10.01	3.31
0.3	3.02	21.96	1.87	7.21	3.17	0.21	12.50	-8.99	11.10	3.72
0.4	2.97	21.83	1.85	7.40	3.52	0.31	12.53	-8.95	12.16	3.77
0.5	2.87	21.79	1.77	7.20	3.62	1.10	12.75	-9.13	13.05	3.78
0.6	2.98	22.72	1.62	7.70	3.72	1.09	12.85	-9.23	14.35	3.97
0.7	2.81	25.80	1.57	7.36	4.03	1.12	12.77	-9.21	15.19	4.00
0.8	2.70	27.33	1.27	7.61	4.39	-0.37	12.76	-9.43	15.50	4.51
0.9	2.66	28.21	1.05	7.56	4.72	-0.22	12.73	-9.57	15.74	4.43
1.0	2.50	30.45	0.68	7.50	4.85	-0.48	12.85	-9.44	16.03	4.58
1.1	2.60	30.37	0.55	7.51	5.12	-0.84	12.51	-9.44	16.56	4.58
1.2	2.36	34.51	-0.10	7.53	5.38	-0.93	12.34	-9.83	17.49	4.54
1.3	2.09	36.29	-0.41	7.45	5.96	-1.15	12.22	-9.78	17.79	4.82
1.4	2.07	38.53	-1.00	7.27	6.40	-1.44	11.84	-9.81	18.03	4.90
1.5	2.01	41.90	-1.19	7.20	6.58	-1.90	11.67	-10.06	18.82	4.77
1.6	1.81	44.52	-1.56	7.25	7.09	-2.25	11.27	-10.41	19.50	4.81
1.7	1.53	48.91	-2.08	7.35	7.32	-2.65	11.06	-10.40	20.28	5.00
1.8	1.64	50.68	-2.61	6.97	7.94	-3.06	10.73	-10.70	21.21	4.97
1.9	1.29	56.36	-3.37	7.20	8.47	-3.66	10.35	-10.96	22.63	5.08
2.0	1.11	59.14	-3.86	7.06	9.00	-4.01	10.09	-11.91	22.90	5.08

Таблица 2

x_i	$y_i = f(x_i)$									
	Номера вариантов									
	11	12	13	14	15	16	17	18	19	20
0.1	8.15	-6.90	0.17	3.30	1.04	0.08	3.09	-0.86	0.00	-0.65
0.3	8.41	-7.01	0.07	2.49	1.47	0.14	3.25	-0.77	-0.09	-1.00
0.5	8.58	-7.19	0.17	3.02	1.78	0.37	3.48	-0.56	-0.26	-0.87
0.7	8.84	-7.11	0.05	3.27	2.01	0.36	3.47	-0.46	-0.20	-0.89
0.9	9.28	-7.31	0.12	3.43	2.19	0.44	3.55	-0.28	-0.29	-0.75
1.1	9.46	7.78	0.00	3.70	2.60	0.48	3.59	-0.24	-0.14	-0.59
1.3	10.02	-7.64	0.01	3.70	2.93	0.27	3.28	-0.36	-0.26	-0.44
1.5	10.11	-7.85	-0.05	3.85	3.22	0.39	3.50	-0.43	-0.45	-0.61
1.7	10.61	-8.18	-0.21	3.89	3.50	0.50	3.61	-0.56	-0.43	-0.17
1.9	11.03	-8.39	-0.50	3.98	4.01	0.48	3.59	-0.59	-0.71	0.13
2.1	11.34	-8.79	-0.50	4.02	4.22	0.69	3.80	-0.70	-0.70	0.53
2.3	11.86	-9.02	-0.86	4.21	4.71	0.50	3.61	-1.01	-1.00	0.67
2.5	12.33	-9.48	-1.24	4.22	5.23	0.31	3.42	-1.03	-1.01	1.00
2.7	12.81	-9.93	-1.47	4.37	5.78	0.37	3.48	-1.47	-1.17	1.34
2.9	13.21	-10.26	-1.79	4.36	6.27	0.43	3.54	-1.68	-1.39	1.49
3.1	13.67	-10.91	-2.25	4.39	6.75	0.33	3.44	-1.93	-1.22	1.81
3.3	14.23	-11.41	-2.55	4.54	7.16	0.31	3.42	-2.28	-1.43	2.37
3.5	14.68	-11.91	-3.18	4.33	7.76	0.09	3.20	-2.53	-1.81	2.72
3.7	15.35	-12.30	-3.60	4.54	8.30	0.08	3.19	-2.93	-1.84	3.03
3.9	15.93	-13.00	-3.93	4.53	9.00	0.03	3.14	-3.07	-1.99	3.51

Таблица 3

x_i	$y_i = f(x_i)$									
	Номера вариантов									
	21	22	23	24	25	26	27	28	29	30
0.15	0.16	1.89	-1.92	1.10	-2.80	0.00	4.01	0.12	4.13	2.97
0.30	0.02	2.07	-1.60	1.20	-2.66	0.01	4.06	0.31	4.11	3.07
0.45	0.28	2.30	-1.57	1.18	-2.36	0.24	3.88	0.48	3.87	3.04
0.60	0.42	2.26	-1.41	1.14	-2.41	0.74	3.98	0.45	3.74	3.30
0.75	0.31	2.34	-1.36	1.17	-2.13	1.02	4.36	0.84	3.85	3.27
0.90	0.41	2.66	-0.97	1.00	-1.82	1.31	4.18	0.73	3.71	3.54
1.05	0.42	2.88	-0.59	0.99	-1.74	1.53	4.16	0.77	3.53	3.79
1.20	0.36	2.85	-0.71	0.95	-1.76	1.90	4.51	0.64	3.56	4.07
1.35	0.45	3.16	-0.15	0.54	-1.64	2.29	4.53	0.74	3.19	4.30
1.50	0.65	3.49	0.01	0.32	-1.46	2.61	4.38	0.53	3.04	4.51
1.65	0.67	3.88	0.22	0.15	-1.30	3.15	4.76	0.28	2.83	4.83
1.80	0.53	4.22	0.63	0.02	-1.27	3.42	4.66	0.24	2.54	5.06
1.95	0.50	4.45	1.07	-0.30	-1.22	3.89	4.82	0.00	2.41	5.40
2.10	0.35	4.99	1.42	-0.40	-1.11	4.58	4.77	0.03	1.97	5.83
2.25	0.35	5.36	1.68	-0.90	-1.02	4.82	5.12	0.35	1.78	6.54
2.40	0.13	5.71	2.49	-1.37	0.89	5.42	5.23	0.46	1.53	6.68
2.55	0.39	6.51	2.57	-1.65	0.89	6.07	5.40	0.88	1.04	7.36
2.70	0.14	7.35	3.09	-2.00	-1.02	6.44	5.84	1.27	0.86	7.91
2.85	0.14	8.02	3.40	-2.42	0.97	7.15	5.86	1.68	0.48	8.39
3.00	0.09	8.96	4.00	-3.13	0.99	7.66	6.01	1.98	-0.09	8.98

3.3. Глобальная полиномиальная интерполяция

Пусть функция $y = f(x)$ интерполируется на отрезке $[a, b]$. **Метод решения этой задачи единственным для всего отрезка многочленом $P_n(x)$ называется глобальной полиномиальной интерполяцией.** Надежда приблизить $y = f(x)$ везде на $[a, b]$ с заданной точностью ε единственным многочленом $P_n(x)$ базируются на теореме Вейерштрасса*.

Теорема 3.1. Пусть функция $y = f(x)$ непрерывна на $[a, b]$. Тогда для любого $\varepsilon > 0$ существует полином $P_n(x)$ степени $n = n(\varepsilon)$ такой, что $\max_{[a, b]} |f(x) - P_n(x)| < \varepsilon$.

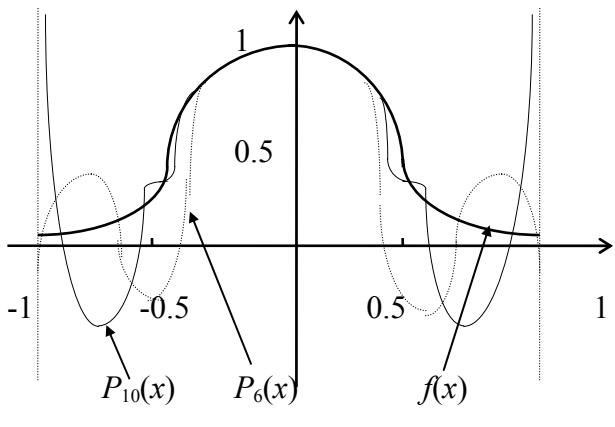
Однако существуют очень веские причины, по которым глобальная интерполяция многочленами высокой степени в вычислительной практике не используется. Обычный подход увеличения точности интерполяции - увеличение числа узлов. Однако не существует единой для всех непрерывных на отрезке $[a, b]$ функций стратегии выбора узлов интерполяции. Чаще всего узлы располагаются на $[a, b]$ равномерно. Но даже для очень гладких функций это иногда не приносит желаемого эффекта.

Классический пример - функция Рунге** $f(x) = \frac{1}{1 + 25x^2}$, $x \in [-1, 1]$. Если использовать глобальную аппроксимацию на $[-1, 1]$ с равномерным распределением узлов, то при больших n интерполяция дает очень хорошие результаты в центральной части отрезка. В то

* Карл Вейерштрасс (1815-1897) – немецкий математик.

** Карл Давид Тольме Рунге (1856-1927) - немецкий физик и математик.

же время последовательность $P_n(x)$ расходится при $n \rightarrow \infty$ для $0.73 < |x| \leq 1$ (см. рисунок слева).



Таким образом, равномерное распределение узлов оказалось неудачным. Итак, не существует единой для всякой функции $y = f(x)$ стратегии выбора узлов интерполяции. Об этом же говорит и теорема Фабера*.

Теорема 3.2. **Какова бы ни была стратегия выбора узлов интерполяции, найдется непрерывная на отрезке $[a, b]$ функция $y = f(x)$, для которой** $\max_{[a, b]} |f(x) - P_n(x)| \rightarrow \infty$ при $n \rightarrow \infty$.

Однако если функция гладкая (непрерывно дифференцируемая), то такая стратегия существует.

Теорема 3.3. **Пусть в качестве узлов интерполяции на отрезке $[a, b]$ выбираются узлы полиномов Чебышева**. Тогда для любой непрерывно дифференцируемой на отрезке $[a, b]$ функции $y = f(x)$ метод интерполяции сходится.**

Практическая реализация стратегии выбора узлов интерполяции возможна и оправдана в довольно редких случаях и просто невозможна тогда, когда приходится иметь дело с заданной таблицей значений функции.

3.4. Чувствительность интерполяционного многочлена к погрешностям входных данных

Помимо погрешности от приближенной замены $y = f(x)$ на $P_n(x)$ возникает еще дополнительная погрешность, связанная с тем, что значения интерполируемой функции тоже задаются с погрешностью. Пусть заданные в узлах x_i , $i = \overline{0, n}$ значения y_i^* содержат погрешности ε_i . Тогда $P_n^*(x) = \sum_{j=0}^n y_j^* l_{nj}(x)$ содержит погрешность $P_n(x) - P_n^*(x) = \sum_{j=0}^n \varepsilon_j l_{nj}(x)$, где $\{l_{nj}(x)\}_{j=0}^n$ - Лагранжев базис.

Пусть известно, что верхняя граница погрешности y_i^* равна $\bar{\Delta}(y^*)$, то есть $|\varepsilon_i| \leq \bar{\Delta}(y^*)$ для $\forall i = \overline{0, n}$. Тогда для верхней границы соответствующей погрешности многочлена $\bar{\Delta}(P_n^*) = \max_{[a, b]} |P_n(x) - P_n^*(x)|$ справедлива оценка

$$\bar{\Delta}(P_n^*(x)) \leq \Lambda_n \bar{\Delta}(y^*), \text{ где } \Lambda_n = \max_{[a, b]} \sum_{j=0}^n |l_{nj}(x)| - \text{константа Лебега}^{***}. \quad (3.4.1)$$

В задаче интерполяции константа Лебега играет роль абсолютного числа обусловленности, то есть в самом неблагоприятном случае погрешность входных данных при интерполяции может возрасти в Λ_n раз. Величина Λ_n зависит от расположения узлов интерполяции. Например, если в качестве узлов интерполяции взяты нули многочленов Чебышева, то

$$\Lambda_n \approx \frac{2}{\pi} \ln(n+1) + 1. \quad (3.4.2)$$

* Жорж Фабер (1877-1966) - швейцарский математик.

** Пафнутий Львович Чебышев (1821-1894) – русский математик и механик.

*** Анри Леон Лебег (1875-1941) - французский математик.

Если же узлы равноотстоящие, то $\Lambda_n > \frac{2^{n-1}}{(2n-1)\sqrt{n}}$ и уже при $n \geq 4$ обусловленность

задачи резко ухудшается. Из этого следует важный практический вывод: в вычислениях не следует использовать интерполяционные многочлены высокой степени с равноотстоящими узлами.

3.5. Многочлены Чебышева

Система функций $\{\varphi_n(x)\}$, заданных на $[a, b]$, называется ортогональной на $[a, b]$, если

1. $\int_a^b \varphi_n^2(x) dx = d_n \neq 0$ и \exists для $\forall n = 0, 1, 2, \dots$
 2. $\int_a^b \varphi_n(x) \varphi_m(x) dx = \begin{cases} 0, & m \neq n, \\ d_n \neq 0, & n = m. \end{cases}$
- (3.5.1)

Система функций $\{\varphi_n(x)\}$, заданных на $[a, b]$, называется ортогональной на $[a, b]$ с весом $\rho(x)$, если

1. $\int_a^b \varphi_n^2(x) \rho(x) dx = d_n \neq 0$ и \exists для $\forall n = 0, 1, 2, \dots$
 2. $\int_a^b \varphi_n(x) \varphi_m(x) \rho(x) dx = \begin{cases} 0, & m \neq n, \\ d_n \neq 0, & n = m. \end{cases}$
- (3.5.2)

Функция $\rho(x)$ называется весовой функцией для системы $\{\varphi_n(x)\}$. Если $d_n = 1$ для $\forall n$, то система функций $\{\varphi_n(x)\}$ называется ортонормированной.

В качестве примера системы функций, ортогональной с весом, приведем многочлены Чебышева, которые известны еще и тем, что являются полиномами, наименее уклоняющимися от нуля. Эти многочлены определяют разными способами. Например:

$$1. T_n(x) = \frac{1}{2} \left[(x + \sqrt{x^2 - 1})^n + (x - \sqrt{x^2 - 1})^n \right], \quad n = 1, 2, 3, \dots \quad (3.5.3)$$

2. Являются решениями следующего дифференциального уравнения:

$$(1 - x^2) T_n''(x) - x T_n'(x) + n^2 T_n(x) = 0. \quad (3.5.4)$$

3. Определяются из формулы Родрига*

$$\frac{T_n(x)}{\sqrt{1-x^2}} = \frac{(-1)^n}{n!} \frac{d^n \left[(1-x^2)^{\frac{n-1}{2}} \right]}{dx^n}, \quad n = 0, 1, 2, \dots \quad (3.5.5)$$

4. Определяются рекуррентно:

$$\begin{aligned} T_0(x) &= 1, \quad T_1(x) = x, \\ T_{n+1}(x) &= 2xT_n(x) - T_{n-1}(x), \quad n = 1, 2, 3, \dots \end{aligned} \quad (3.5.6)$$

Иногда в качестве полиномов Чебышева берут функции $\tilde{T}_n(x) = \frac{1}{2^n} T_n(x)$.

$$5. T_n(x) = \cos(n \cdot \arccos x), \quad n = 0, 1, 2, \dots \quad x \in [-1, 1]. \quad (3.5.7)$$

$$6. T_n(x) = \frac{n}{2} \sum_{m=0}^{\left[\frac{n}{2}\right]} (-1)^m \frac{(n-m-1)!}{m!(n-2m)!} (2x)^{n-2m}. \quad (3.5.8)$$

Многочлены Чебышева обладают множеством замечательных свойств.

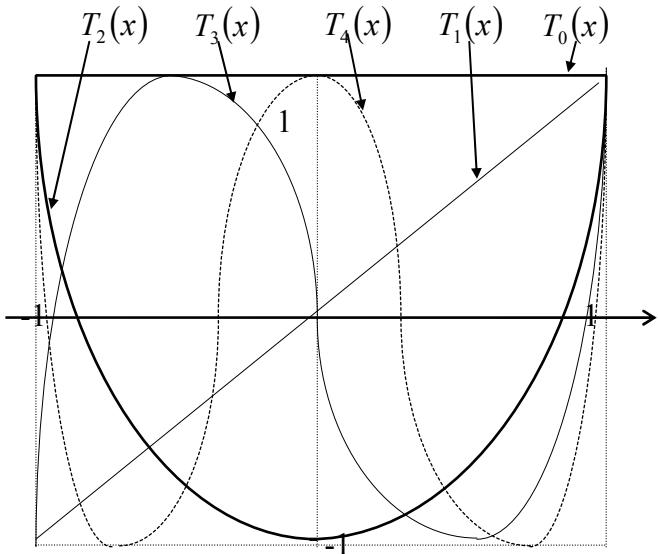
* Родриг; Бенжамен Оленд Родригес (1794-1851) - французский математик и экономист.

Теорема 3.4. Полиномы Чебышева $T_n(x)$ образуют на отрезке $[-1, 1]$ ортогональную систему с весом $\rho(x) = \frac{1}{\sqrt{1-x^2}}$, то есть

$$\int_{-1}^1 T_n(x)T_m(x) \frac{dx}{\sqrt{1-x^2}} = \begin{cases} 0, & n \neq m, \\ \pi, & n = m = 0, \\ \frac{\pi}{2}, & n = m > 0. \end{cases} \quad (3.5.9)$$

Действительно, $\int_{-1}^1 T_n(x)T_m(x) \frac{dx}{\sqrt{1-x^2}} = \left\langle \begin{array}{l} x = \cos t, \\ 0 \leq t \leq \pi, \\ T_n(\cos t) = \cos(n \cdot \arccos(\cos t)) = \cos nt \end{array} \right\rangle =$

$$= \int_{-\pi}^0 \cos nt \cdot \cos mt \frac{(-\sin t)}{|\sin t|} dt = \frac{1}{2} \int_{-\pi}^{\pi} \cos nt \cdot \cos mt dt = \begin{cases} 0, & n \neq m, \\ \pi, & n = m, \\ \frac{\pi}{2}, & n = m > 0. \end{cases}$$



$$\begin{aligned} T_0(x) &= 1, \\ T_1(x) &= x, \\ T_2(x) &= 2x^2 - 1, \\ T_3(x) &= 4x^3 - 3x, \\ T_4(x) &= 8x^4 - 8x^2 + 1, \\ T_5(x) &= 16x^5 - 20x^3 + 5x \end{aligned}$$

и так далее.

Теорема 3.5. При четном n многочлен $T_n(x)$ содержит только четные степени x и является четной функцией, а при нечетном n многочлен $T_n(x)$ содержит только нечетные степени x и является нечетной функцией.

Теорема 3.6. При $n \geq 1$ старший коэффициент многочлена $T_n(x)$ равен 2^{n-1} , то есть $T_n(x) = 2^{n-1}x^n + \dots$

Теорема 3.7. При $n \geq 1$ многочлен $T_n(x)$ имеет ровно n действительных корней, расположенных на отрезке $[-1, 1]$ и вычисляемых по формуле

$$x_k = \cos \frac{(2k+1)\pi}{2n}, \quad k = 0, 1, 2, \dots, n-1. \quad (3.5.10)$$

Теорема 3.8. При $n \geq 0$ справедливо равенство $\max_{[-1, 1]} |T_n(x)| = 1$. Если $n \geq 1$, то этот максимум достигается ровно в $n+1$ точках, которые находятся по формуле

$$x_m = \cos \frac{\pi m}{n}, \quad m = 0, 1, \dots, n. \quad (3.5.11)$$

При этом $T_n(x_m) = (-1)^m$, то есть максимумы и минимумы многочлена Чебышева чередуются.

Теоремы 3.7 и 3.8 легко доказываются с помощью формулы (3.5.7).

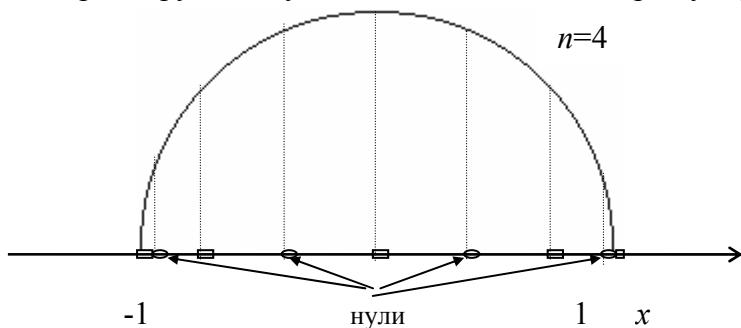
Назовем величину $\max_{[-1,1]} |P_n(x)|$ уклонением многочлена $T_n(x)$ от нуля. Тогда справедлива следующая, доказанная П.Л. Чебышевым в 1854 г. теорема.

Теорема 3.9. Среди всех многочленов фиксированной степени $n \geq 1$ со старшим коэффициентом a_n , равным единице, наименьшее уклонение от нуля, равное 2^{1-n} , имеет многочлен $\tilde{T}_n(x) = \frac{1}{2^{n-1}} T_n(x)$.

Последнее свойство имеет особую ценность для приложений. Действительно, тогда для любого многочлена $P_n(x) = x^n + a_{n-1}x^{n-1} + \dots + a_0$, отличного от $\tilde{T}_n(x)$, справедливо неравенство

$$\frac{1}{2^{n-1}} = \max_{[-1,1]} |\tilde{T}_n(x)| < \max_{[-1,1]} |P_n(x)|. \quad (3.5.12)$$

В силу формул (3.5.10) и (3.5.11) нули и точки экстремума полинома $T_n(x)$ можно построить следующим образом: разделив полуокружность, опирающуюся на отрезок $[-1, 1]$ на $2n$ частей, спроектируем полученные точки на диаметр. Нумеруя проекции слева направо, получим,



что все проекции с нечетными номерами являются нулями полинома $T_n(x)$, а с четными – его точками экстремума. Из геометрических соображений вытекает, что как нули, так и точки экстремума полинома $T_n(x)$ сгущаются к концам отрезка $[-1, 1]$.

3.6. Решение задачи минимизации оценки погрешности

Предположим, что значение заданной на отрезке $[a, b]$ функции $f(x)$ можно вычислить в произвольной точке x . Однако по некоторым причинам целесообразнее заменить прямое вычисление функции $f(x)$ вычислением значений ее интерполяционного многочлена. Для такой замены необходимо один раз получить таблицу значений $f(x)$ в выбранных точках $x_0, x_1, \dots, x_n \in [a, b]$. При этом естественно стремиться к такому выбору узлов интерполяции, который позволит сделать минимальной величину $\Delta(P_n(x)) = \max_{[a,b]} |f(x) - P_n(x)|$.

Пусть о функции $f(x)$ известно лишь то, что она непрерывно дифференцируема $n+1$ раз на отрезке $[a, b]$. Тогда

$$\bar{\Delta}(P_n(x)) = \frac{M_{n+1}}{(n+1)!} \max_{[a,b]} |\omega_{n+1}(x)|. \quad (3.6.1)$$

Найдем теперь набор узлов интерполяции x_0, x_1, \dots, x_n , при котором $\bar{\Delta}(P_n(x)) \rightarrow \min$. Пусть сначала $[a, b] = [-1, 1]$. В этом случае величина (3.6.1) будет минимальна, если будет минимальна $\max_{[-1,1]} |\omega_{n+1}(x)|$. Но этим свойством обладают полиномы Чебышева, следовательно,

$\omega_{n+1}(x) \equiv \tilde{T}_{n+1}(x)$, и набор узлов определен $x_k = \cos \frac{2k+1}{2(n+1)}\pi$, $k = 0, 1, \dots, n$. Это нули многочлена $\tilde{T}_{n+1}(x)$. При таком выборе

$$\bar{\Delta}(P_n(x)) = \frac{M_{n+1}}{2^n(n+1)!}, \quad (3.6.2)$$

причем любой другой выбор узлов дает большее значение верхней границы погрешности. Для формулы Тейлора, например, $\bar{\Delta}(P_n(x)) = \frac{M_{n+1}}{(n+1)!}$, то есть в 2^n раз хуже.

Перейдем теперь к отрезку $[a, b]$. Его можно преобразовать к стандартному отрезку $[-1, 1]$ следующей заменой: $x = \frac{a+b}{2} + \frac{b-a}{2}t$, где $t \in [-1, 1]$. В этом случае

$$\begin{aligned} \omega_{n+1}(x) &= \left[\frac{b-a}{2} \right]^{n+1} \tilde{\omega}_{n+1}(t), \\ \text{где } \tilde{\omega}_{n+1}(t) &= (t - t_0)(t - t_1) \dots (t - t_n) \\ \text{и } x_k &= \frac{a+b}{2} + \frac{b-a}{2}t_k, \quad k = 0, 1, \dots, n. \end{aligned} \quad (3.6.3)$$

Тогда

$$\bar{\Delta}(P_n(x)) = \frac{M_{n+1}}{(n+1)!} \left[\frac{b-a}{2} \right]^{n+1} \max_{[-1, 1]} |\tilde{\omega}_{n+1}(t)| \quad (3.6.4)$$

и минимум этой величины достигается при значениях t_0, t_1, \dots, t_n , совпадающих с нулями многочлена $\tilde{T}_{n+1}(t)$. Таким образом, решение поставленной задачи дает выбор узлов

$$x_k = \frac{a+b}{2} + \frac{b-a}{2} \cos \left(\frac{2k+1}{2n+2} \pi \right), \quad k = 0, 1, \dots, n, \quad (3.6.5)$$

и ему соответствует минимальное значение верхней границы погрешности интерполяции, равное:

$$\bar{\Delta}(P_n(x)) = \frac{M_{n+1}}{2^n(n+1)!} \left[\frac{b-a}{2} \right]^{n+1}. \quad (3.6.6)$$

3.7. Ряд Фурье^{*} по многочленам Чебышева

Функциональный ряд вида $f(x) = \sum_{i=0}^{\infty} c_i \varphi_i(x)$, где $\{\varphi_i(x)\}$ - система базисных функций, называется рядом Фурье функции $f(x)$ на отрезке $[a, b]$ по системе функций $\varphi_i(x)$, $i = 0, 1, \dots$ с весом $\rho(x)$ на $[a, b]$, если коэффициенты c_i вычисляются по формулам

$$\text{вида} \quad c_i = \frac{1}{\int_a^b \varphi_i^2(x) \rho(x) dx} \int_a^b f(x) \varphi_i(x) \rho(x) dx, \quad i = 0, 1, \dots \quad (3.7.1)$$

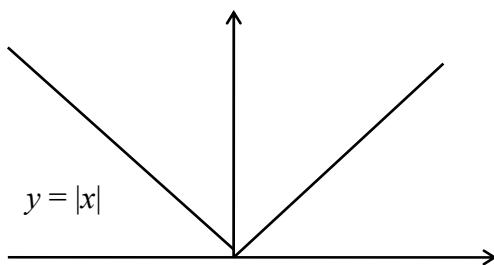
Так как для полиномов Чебышева весовая функция равна: $\rho(x) = \frac{1}{\sqrt{1-x^2}}$ на $[-1, 1]$,

$$\text{то} \quad \begin{cases} f(x) = \sum_{i=0}^{\infty} a_i T_i(x), \text{ где} \\ a_0 = \frac{1}{\pi} \int_{-1}^1 \frac{f(x)}{\sqrt{1-x^2}} dx, \quad a_i = \frac{2}{\pi} \int_{-1}^1 \frac{f(x) T_i(x)}{\sqrt{1-x^2}} dx, \quad i = 1, 2, \dots \end{cases} \quad (3.7.2)$$

* Жан Батист Жозеф Фурье (1768-1830) - французский математик.

Пример. Разложить в ряд Фурье по полиномам Чебышева функцию $y = |x|$ на отрезке $[-1, 1]$.

Функция $y = |x|$ - четная, поэтому, как во всяком ряду Фурье, нечетные коэффициенты будут равны нулю, четные же можно сдвоить, уменьшив при этом интервал интегрирования вдвое. Тогда $a_0 = \frac{2}{\pi} \int_0^1 \frac{f(x)}{\sqrt{1-x^2}} dx$, $a_i = \frac{4}{\pi} \int_0^1 \frac{f(x)T_i(x)}{\sqrt{1-x^2}} dx$. Вычислим коэффициенты разложения:



$$a_0 = \frac{2}{\pi} \int_0^1 \frac{x dx}{\sqrt{1-x^2}} = \begin{cases} x = \cos t, & dx = -\sin t dt, \\ \sqrt{1-x^2} = |\sin t|, & x = 0, t = \frac{\pi}{2}, \\ x = 1, t = 0. & \end{cases} =$$

$$= \frac{2}{\pi} \int_0^{\frac{\pi}{2}} \cos t dt = \frac{2}{\pi} \sin t \Big|_0^{\frac{\pi}{2}} = \frac{2}{\pi}.$$

$$\begin{aligned} a_i &= \frac{4}{\pi} \int_0^1 \frac{x T_i(x) dx}{\sqrt{1-x^2}} = -\frac{4}{\pi} \int_{-\frac{\pi}{2}}^0 \frac{\cos t \sin t \cos it}{\sin t} dt = \frac{4}{\pi} \int_0^{\frac{\pi}{2}} \cos t \cos it dt = \\ &= \frac{2}{\pi} \int_0^{\frac{\pi}{2}} [\cos(t(i-1)) + \cos(t(i+1))] dt = \frac{2}{\pi} \left[\frac{1}{i-1} \sin(t(i-1)) \Big|_0^{\frac{\pi}{2}} + \frac{1}{i+1} \sin(t(i+1)) \Big|_0^{\frac{\pi}{2}} \right] = \\ &= \frac{2}{\pi} \left[\frac{-1}{i-1} \cos \frac{\pi i}{2} + \frac{1}{i+1} \cos \frac{\pi i}{2} \right] = -\frac{4}{\pi(i^2-1)} \cos \frac{\pi i}{2} = \begin{cases} 0, & i = 2m+1, \\ \frac{4}{\pi(4m^2-1)} (-1)^{m+1}, & i = 2m. \end{cases} \end{aligned}$$

Итак, $|x| = \frac{2}{\pi} + \frac{4}{\pi} \sum_{m=1}^{\infty} \frac{(-1)^{m+1}}{(4m^2-1)} T_{2m}(x)$.

Пример. По равномерной (а) и специальной (б) таблице значений функции $y = \lg x$ найти $\lg 0.5$ и оценить погрешность. Использовать формулы Ньютона с разделенными разностями.

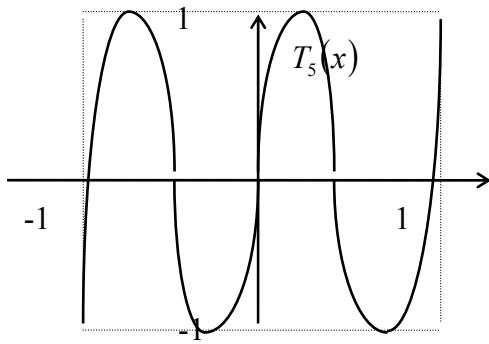
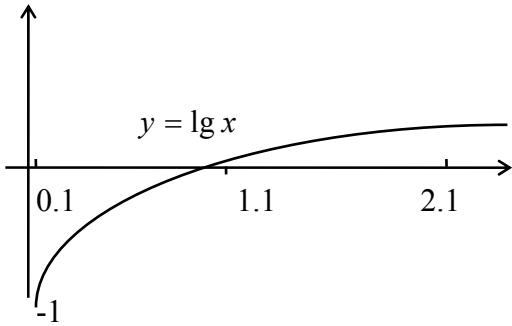
(а)

x	0.1	0.6	1.1	1.6	2.1
y	-1.000000	-0.221849	0.041393	0.204120	0.322219

(б)

x	0.148944	0.512215	1.1	1.687785	2.051057
y	-0.826977	-0.290548	0.041393	0.227317	0.311978

Так как таблицы короткие, то по ним можно вычислить разделенные разности лишь до четвертого порядка. Точки аргумента в таблице (б) - это нули многочлена Чебышева пятой степени, то есть точки, где $T_5(x) = 0$, приведенные к отрезку $[0.1, 2.1]$ по формуле



$$x_k = \frac{a+b}{2} + \frac{b-a}{2} \cos\left(\frac{2k+1}{2n+2}\pi\right), \quad k = 0, 1, 2, 3, 4, \quad n = 4. \quad \text{Действительно, } a = 0.1, b = 2.1,$$

$$k = 0, \quad x_0 = 1.1 + \cos\frac{\pi}{10} = 2.051057,$$

$$k = 1, \quad x_1 = 1.1 + \cos\frac{3\pi}{10} = 1.687785,$$

$$k = 2, \quad x_2 = 1.1 + \cos\frac{\pi}{2} = 1.100000,$$

$$k = 3, \quad x_3 = 1.1 + \cos\frac{7\pi}{10} = 0.512215,$$

$$k = 4, \quad x_4 = 1.1 + \cos\frac{9\pi}{10} = 0.148944.$$

Согласно теории, в случае расположения узлов интерполяции в нулях многочлена Чебышева гарантирована минимальная погрешность интерполяции, равная $\overline{\Delta}(P_n(x)) = \frac{M_{n+1}}{2^n(n+1)!} \left[\frac{b-a}{2} \right]^{n+1}$, в отличие от обычной интерполяции по формуле Ньютона с

разделенными разностями, где погрешность равна: $\overline{\Delta}(P_n(x)) = \frac{M_{n+1}}{(n+1)!} \max_{[a, b]} |\omega_{n+1}(x)|$.

Построим для случаев (а) и (б) таблицы разделенных разностей.

(а)

i	x	y	$f(x_0; x_1)$	$f(x_0; x_1; x_2)$	$f(x_0; x_1; x_2; x_3)$	$f(x_0; x_1; \dots; x_4)$
0	0.1	-1.000000				
			1.556302			
1	0.6	-0.221849		-1.029818		
			0.526484		0.552525	
2	1.1	0.041393		-0.201030		-0.239005
			0.325454		0.074516	
3	1.6	0.204120		-0.089256		
			0.236198			
4	2.1	0.322219				

(6)

i	x	y	$f(x_0; x_1)$	$f(x_0; x_1; x_2)$	$f(x_0; x_1; x_2; x_3)$	$f(x_0; x_1; \dots; x_4)$
0	0.148944	-0.826977				
			1.476663			
1	0.512215	-0.290548		-0.958862		
			0.564732		0.485784	
2	1.100000	0.041393		-0.211318		-0.213107
			0.316313		0.080431	
3	1.687785	0.227317		-0.087547		
			0.233051			
4	2.051057	0.311978				

Вычисления по этим таблицам дают:

(a)

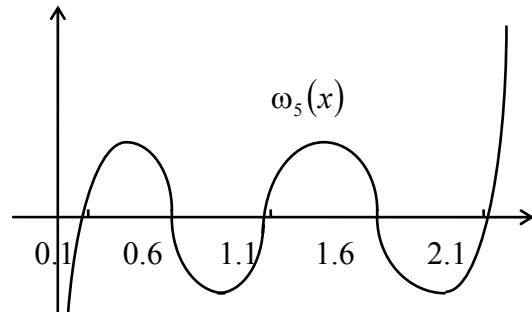
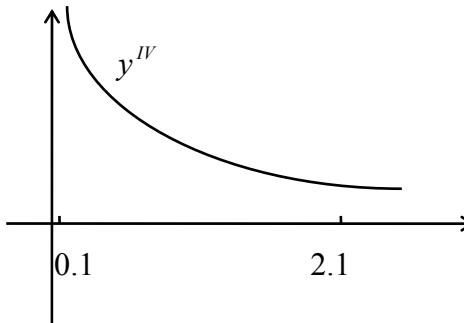
$$\begin{aligned} \lg 0.5 = & -1.000000 + 1.556302 \cdot 0.4 - 1.029818 \cdot 0.4 \cdot (-0.1) + 0.552525 \cdot 0.4 \cdot (-0.1) \cdot (-0.6) - \\ & - 0.239005 \cdot 0.4 \cdot (-0.1) \cdot (-0.6) \cdot (-1.1) = -1.000000 + 0.622521 + 0.041193 + 0.013261 + \\ & + 0.006310 = -0.316716. \end{aligned}$$

(б)

$$\begin{aligned} \lg 0.5 = & -0.826977 + 1.476663 \cdot 0.351056 - 0.958862 \cdot 0.351056 \cdot (-0.012215) + 0.485784 \cdot \\ & \cdot 0.351056 \cdot (-0.012215) \cdot (-0.6) - 0.213107 \cdot 0.351056 \cdot (-0.012215) \cdot (-0.6) \cdot (-1.187785) = \\ & = -0.826977 + 0.518391 + 0.004112 + 0.001250 + 0.000651 = -0.302573. \end{aligned}$$

Истинное значение $\lg 0.5 = -0.301030$, таким образом, предельная погрешность в случае (а) достигает $\bar{\Delta}_a = 0.01569$, а в случае (б) $\bar{\Delta}_b = 0.00154$, то есть на порядок меньше.

Так как таблица разностей очень короткая, а $y = \lg x$ вблизи точки $x = 0.1$ меняется быстро (см. график функции $y = \lg x$), то попытка оценить погрешность по известным формулам не даст достоверного результата, то есть погрешность будет слишком завышена. Действительно, $y = \lg x$, $y^{IV} = \frac{24}{x^5} \lg e$, $\max_{[0.1, 2.1]} |y^{IV}| > 1 \cdot 10^6$. Так же, огрубляя оценку и вспоминая график



функции $\omega_5(x)$, будем иметь $\max_{[0.1, 2.1]} |\omega_5(x)| < h_{\max}^5$, где $h_{\max} = \max_{0 \leq i \leq 4} h_i$. Тогда

$R_a \leq \frac{1 \cdot 10^6 \cdot 0.03125}{120} = 260$. Аналогично для случая (б) $R_b \leq \frac{1.4 \cdot 10^5 \cdot 1.61051}{120 \cdot 32} = 58$. Оба результата не реальны.

3.8. Лабораторная работа №5. Экономизация степенных рядов

Пусть функция $f(x)$ непрерывна на отрезке $[a, b]$. Говорят, что многочлен $P_n(x)$ приближает функцию $f(x)$ равномерно на отрезке $[a, b]$ с точностью ε , если

$$\Delta(P_n) = \max_{[a, b]} |f(x) - P_n(x)| \leq \varepsilon. \quad (3.8.1)$$

Можно поставить и решить задачу нахождения многочлена $Q_n(x)$, для которого величина погрешности равномерного приближения ε минимальна. Эта задача называется задачей о наилучшем равномерном приближении. Теоретически доказано, что для любой непрерывной на $[a, b]$ функции многочлен наилучшего равномерного приближения степени n существует и он единственный.

В большинстве практических случаев задача о наилучшем равномерном приближении функции $f(x)$ очень трудна и требует применения специально разработанных численных методов. Если же ограничиться заданной точностью ε , то можно построить многочлен, близкий к наилучшему в следующих двух случаях.

1. Если для функции $f(x)$ ее $n+1$ -я производная $f^{(n+1)}(x)$ медленно меняется на отрезке $[a, b]$, то интерполяционный многочлен $P_n(x)$ с узлами, расположенными в нулях многочлена Чебышева $T_n(x)$, близок к многочлену наилучшего равномерного приближения (см. пример в подразд. 3.7).

2. Если функция $f(x)$ представлена на отрезке $[a, b]$ равномерно сходящимся степенным рядом

$$f(x) = \sum_{k=0}^{\infty} a_k x^k, \quad (3.8.2)$$

то на основе этого ряда можно построить многочлен минимальной степени, равномерно приближающий функцию $f(x)$ на отрезке $[a, b]$ с заданной точностью ε .

Рассмотрим вторую задачу более подробно. Метод ее решения называется экономизацией степенных рядов.

Пусть $P_n(x) = a_n x^n + a_{n-1} x^{n-1} + \dots + a_0$ - многочлен степени $n \geq 1$, представляющий собой отрезок ряда (3.8.2) на стандартном отрезке $[-1, 1]$. Аппроксимируем этот многочлен многочленом на единицу меньшей степени $Q_{n-1}(x)$ наилучшего равномерного приближения. Их разность многочлен $R_n(x) = P_n(x) - Q_{n-1}(x)$ должен удовлетворять условию

$$\max_{[-1, 1]} |R_n(x)| \Rightarrow \min. \quad (3.8.3)$$

Как показано в подразд. 3.6, решением этой задачи является многочлен $R_n(x) = \frac{a_n}{2^{n-1}} T_n(x)$, для которого $\max_{[-1, 1]} |R_n(x)| = \frac{|a_n|}{2^{n-1}}$. Тогда многочлен наилучшего равномерного приближения имеет вид $Q_{n-1}(x) = P_n(x) - \frac{a_n}{2^{n-1}} T_n(x)$. (3.8.4)

Практически сначала берут отрезок ряда (3.8.2) $P_n(x) = \sum_{k=0}^n a_k x^k$, аппроксимирующий функцию $f(x)$ с точностью $\varepsilon_0 < \varepsilon$. Далее степень многочлена понижают на единицу по описанной процедуре. Если погрешность понижения степени $P_n(x)$, равная $\varepsilon_1 = \frac{|a_n|}{2^{n-1}}$, такова, что $\varepsilon_0 + \varepsilon_1 < \varepsilon$, то многочлен $P_n(x)$ заменяется многочленом (3.8.4). Затем степень $Q_{n-1}(x)$ снова

понижают и так до тех пор, пока $\varepsilon_0 + \varepsilon_1 + \varepsilon_2 + \dots + \varepsilon_i < \varepsilon$. Процесс останавливают, когда очередное $i+1$ -е понижение степени дает $\varepsilon_0 + \varepsilon_1 + \varepsilon_2 + \dots + \varepsilon_{i+1} > \varepsilon$. В этом случае $f(x) \approx Q_{n-i}(x)$.

Пример. Найти многочлен минимальной степени, аппроксимирующий интеграл вероятности $erf(x) = \frac{2}{\sqrt{\pi}} \int_0^x e^{-t^2} dt$ на отрезке $[-1, 1]$ с точностью $\varepsilon = 1.5 \cdot 10^{-3}$.

$$\text{Интегралом вероятностей называют функцию } y = erf(x) = \frac{2}{\sqrt{\pi}} \int_0^x e^{-t^2} dt = \\ = \frac{2x}{\sqrt{\pi}} \left(1 - \frac{x^2}{1! \cdot 3} + \frac{x^4}{2! \cdot 5} - \frac{x^6}{3! \cdot 7} + \dots \right) = \frac{2x}{\sqrt{\pi}} \sum_{k=1}^{\infty} \frac{x^{2k-2}}{(k-1)! \cdot (2k-1)}, |x| < \infty. \quad (3.8.5)$$

Это знакопеременный ряд, поэтому погрешность приближения по признаку Лейбница оценивается величиной ε_0 , равной модулю первого отброшенного члена ряда.

Выберем $k=5$, то есть оставим в ряду (3.8.5) пять членов. Тогда если $|x| \leq 1$, то

$$\varepsilon_0 = \frac{2}{5! \cdot 11 \cdot \sqrt{\pi}} \approx 8.5 \cdot 10^{-4}. \text{ Таким образом, многочлен } P_9(x) = \frac{2x}{\sqrt{\pi}} \left(1 - \frac{x^2}{1! \cdot 3} + \frac{x^4}{2! \cdot 5} - \frac{x^6}{3! \cdot 7} + \frac{x^8}{4! \cdot 9} \right)$$

аппроксимирует функцию $y = erf(x)$ с точностью $\varepsilon_0 \approx 8.5 \cdot 10^{-4}$.

Понизим степень этого многочлена на единицу по формуле (3.8.4):

$Q_8(x) = P_9(x) - \frac{a_9}{2^8} T_9(x) = \frac{2x}{\sqrt{\pi}} - \frac{2x^3}{3\sqrt{\pi}} + \frac{2x^5}{10\sqrt{\pi}} - \frac{2x^7}{42\sqrt{\pi}} + \frac{2x^9}{216\sqrt{\pi}} - \frac{2}{216\sqrt{\pi} \cdot 2^8} (256x^9 - 576x^7 + 432x^5 - 120x^3 + 9x)$. Здесь a^9 - коэффициент при x^9 в формуле (3.8.5), а многочлен Чебышева $T_9(x)$ посчитан по рекуррентной формуле (3.5.6). Поскольку и исходный многочлен $P_9(x)$, и многочлен Чебышева $T_9(x)$ содержат лишь нечетные степени аргумента, то в результате проведенной процедуры степень исходного многочлена понизится на две единицы. В результате получим

$$Q_7(x) = \frac{1}{\sqrt{\pi}} \left(\frac{6143}{3072} x - \frac{763}{1152} x^3 + \frac{59}{320} x^5 - \frac{3}{112} x^7 \right). \quad (3.8.6)$$

Понижение степени многочлена $P_9(x)$ сопровождается дополнительной погрешностью (3.8.3) $\varepsilon_1 = \frac{|a_9|}{2^8} = \frac{2}{216\sqrt{\pi} \cdot 2^8} \approx \frac{1}{49005} = 0.2 \cdot 10^{-4}$. Тогда $\varepsilon_0 + \varepsilon_1 = 8.7 \cdot 10^{-4} < \varepsilon$, то есть многочлен наилучшего равномерного приближения $Q_7(x)$ обеспечивает заданную точность аппроксимации.

Возможно ли дальнейшее понижение степени? При этом величина дополнительной погрешности будет $\varepsilon_2 = \frac{|a'_7|}{2^6} = \frac{3}{112\sqrt{\pi} \cdot 2^6} \approx 2.3 \cdot 10^{-4}$. Суммарная погрешность возрастет до $\varepsilon_0 + \varepsilon_1 + \varepsilon_2 = 1.1 \cdot 10^{-3} < \varepsilon$, поэтому возможно дальнейшее понижение степени многочлена $Q_7(x)$.

Итак, по формуле (3.8.4)

$$Q_5(x) = Q_7(x) - \frac{a'_7}{2^6} T_7(x) = \frac{1}{\sqrt{\pi}} \left(\frac{6143}{3072} x - \frac{763}{1152} x^3 + \frac{59}{320} x^5 - \frac{3}{112} x^7 \right) + \frac{3}{112\sqrt{\pi} \cdot 2^6} \times \\ \times (64x^7 - 112x^5 + 56x^3 - 7x) = \frac{1}{\sqrt{\pi}} \left(\frac{3067}{1536} x - \frac{23}{36} x^3 + \frac{11}{80} x^5 \right) = \\ = 1.126543x - 0.360454x^3 + 0.077576x^5. \quad (3.8.7)$$

Дальнейшее понижение степени многочлена наилучшего равномерного приближения в пределах заданной точности $\varepsilon = 1.5 \cdot 10^{-3}$ невозможно, ибо ошибка при таком понижении составит: $\varepsilon_3 = \frac{11}{80\sqrt{\pi} \cdot 2^4} \approx 4.8 \cdot 10^{-3}$. Тогда $\varepsilon_0 + \varepsilon_1 + \varepsilon_2 + \varepsilon_3 \approx 6.0 \cdot 10^{-3} > \varepsilon$. Поэтому решением задачи является многочлен (3.8.7).

Выполним теперь эти же действия средствами пакета Mathcad, при этом, как и в лабораторной работе №2, воспользуемся меню Symbolics (Символьные вычисления). Это меню содержит операции символьной математики. Пункт Evaluate (Вычислить) содержит три операции: Symbolically (Символьно), Floating Point (С плавающей запятой) и Complex (Комплексное).

Строчкой Simplify (Упростить) мы уже пользовались во второй лабораторной работе. Совершенно аналогично после щелчка по Expand (Развернуть) в выделенном выражении раскрываются скобки, после щелчка по кнопке Factor (Разложить на множители) выделенное выражение раскладывается на множители, наконец, результатом щелчка по кнопке Collect (Собрать) является приведение подобных членов в выделенном выражении.

Итак, наберем в рабочем документе

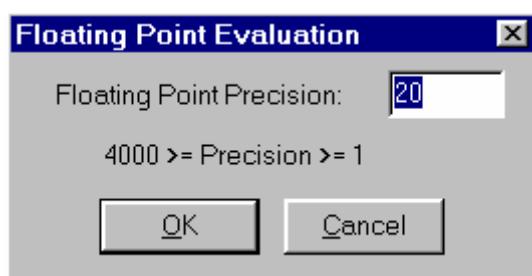
$ORIGIN := 1$

$$P9(x) := 2 \cdot \frac{x}{\sqrt{\pi}} \cdot \left(1 - \frac{x^2}{3 \cdot 1} + \frac{x^4}{5 \cdot 2} - \frac{x^6}{7 \cdot 6} + \frac{x^8}{9 \cdot 24} \right)$$

После щелчка по кнопкам Symbolics → Expand ниже определенной функции появится выражение

$$2 \cdot \frac{x}{\sqrt{\pi}} - \frac{2}{3} \cdot \frac{x^3}{\sqrt{\pi}} + \frac{1}{5} \cdot \frac{x^5}{\sqrt{\pi}} - \frac{1}{21} \cdot \frac{x^7}{\sqrt{\pi}} + \frac{1}{108} \cdot \frac{x^9}{\sqrt{\pi}}.$$

Числовые коэффициенты этого выражения можно представить в десятичной форме. Для этого выберем операции с плавающей запятой: Symbolics → Evaluate → Floating Point. Появится дополнительное меню:



Зададим в маленьком окошечке этого меню, например, девять значащих чисел после запятой. После щелчка по кнопке OK в рабочем документе появится выражение исходного многочлена в виде

$$1.12837917 \cdot x - 0.37612639 \cdot x^3 + 0.112837917 \cdot x^5 - 0.0268661707 \cdot x^7 + 0.00522397763.$$

Для вычисления многочленов Чебышева в пакете Mathcad имеются две функции: **Tcheb(n,x)** и **Ucheb(n,x)**. К сожалению, они становятся встроенными в Mathcad лишь после подгрузки соответствующего электронного учебника. В меню Insert (Вставить) имеется кнопка подменю Function (Функция), которая открывает окно диалога списка встроенных функций пакета. В этом списке **Tcheb** и **Ucheb** отсутствуют.

Однако многочлены Чебышева определяются разными способами (см. формулы (3.5.3) – (3.5.8)). Среди них есть очень простые и легко программируемые определения. Воспользуемся, например, формулами (3.5.3) и (3.5.8).

Введем с клавиатуры

$$T9(x) := \frac{1}{2} \cdot \left[\left(x + \sqrt{x^2 - 1} \right)^9 + \left(x - \sqrt{x^2 - 1} \right)^9 \right]$$

Выделим введенное выражение правым нижним синим углом, тогда после нажатия клавиш Symbolics→Expand получим

$$256 \cdot x^9 - 576 \cdot x^7 + 432 \cdot x^5 - 120 \cdot x^3 + 9 \cdot x$$

Далее

$$T99(x) := \frac{9}{2} \cdot \sum_{m=0}^4 (-1)^m \cdot \frac{\Gamma(9-m)}{\Gamma(m+1) \cdot \Gamma(9-2 \cdot m+1)} \cdot (2 \cdot x)^{9-2 \cdot m}$$

Здесь $\Gamma(n+1) = n!$ -гамма-функция, имеющаяся в окне диалога списка встроенных функций под именем Gamma. Аналогично, после нажатия клавиш Symbolics→Evaluate →Symbolically будет получено то же выражение:

$$256 \cdot x^9 - 576 \cdot x^7 + 432 \cdot x^5 - 120 \cdot x^3 + 9 \cdot x$$

Выполним следующие действия лабораторной работы:

$$\varepsilon_0 := \frac{2}{\sqrt{\pi} \cdot 11 \cdot \Gamma(6)} \quad \varepsilon_0 = 8.548 \cdot 10^{-4} \quad \varepsilon_1 := \frac{2}{\sqrt{\pi} \cdot 216 \cdot 2^8} \quad \varepsilon_1 = 2.041 \cdot 10^{-5}$$

$$Q7(x) := P9(x) - \varepsilon_1 \cdot T9(x)$$

Symbolics→Expand

$$\frac{6143}{3072} \cdot \frac{x}{\sqrt{\pi}} - \frac{763}{1152} \cdot \frac{x^3}{\sqrt{\pi}} + \frac{59}{320} \cdot \frac{x^5}{\sqrt{\pi}} - \frac{3}{112} \cdot \frac{x^7}{\sqrt{\pi}}$$

Symbolics→Evaluate→Floating Point

$$1.12819551 \cdot x - 0.37367765 \cdot x^2 + 0.104022455 \cdot x^5 - 0.015112221 \cdot x^7$$

$$\alpha_1 := \varepsilon_0 + \varepsilon_1 \quad \alpha_1 = 8.752 \cdot 10^{-4} \quad \varepsilon_2 := \frac{3}{\sqrt{\pi} \cdot 112 \cdot 2^6} \quad \varepsilon_2 = 2.361 \cdot 10^{-4}$$

$$\alpha_2 := \alpha_1 + \varepsilon_2 \quad \alpha_2 = 1.111 \cdot 10^{-3}$$

$$T7(x) := \frac{1}{2} \cdot \left[\left(x + \sqrt{x^2 - 1} \right)^7 + \left(x - \sqrt{x^2 - 1} \right)^7 \right]$$

Symbolics→Expand

$$64 \cdot x^7 - 112 \cdot x^5 + 56 \cdot x^3 - 7 \cdot x$$

$$Q5(x) := Q7(x) + \varepsilon_2 \cdot T7(x)$$

Symbolics→Expand

$$\frac{3067}{1536} \cdot \frac{x}{\sqrt{\pi}} - \frac{23}{36} \cdot \frac{x^3}{\sqrt{\pi}} + \frac{11}{80} \cdot \frac{x^5}{\sqrt{\pi}}$$

Symbolics→Evaluate→Floating Point

$$1.12654261 \cdot x - 0.360454457 \cdot x^3 + 0.0775760678 \cdot x^5$$

$$\varepsilon_3 := \frac{11}{80 \cdot \sqrt{\pi} \cdot 2^4} \quad \varepsilon_3 = 4.849 \cdot 10^{-3}$$

$$\alpha_3 := \alpha_2 + \varepsilon_3 \quad \alpha_3 = 5.96 \cdot 10^{-3}$$

Задание №1. Для приведенных функций $f(x)$ построить многочлен наилучшего равномерного приближения на отрезке $[-1, 1]$, выбрав число членов исходного аппроксимирующего полинома по заданной точности ε :

$$1. f(x) = \sin x = \sum_{k=0}^{\infty} (-1)^k \frac{x^{2k+1}}{(2k+1)!}, |x| < \infty, \varepsilon = 10^{-3}.$$

$$2. f(x) = \cos x = \sum_{k=0}^{\infty} (-1)^k \frac{x^{2k}}{(2k)!}, |x| < \infty, \varepsilon = 10^{-3}.$$

$$3. f(x) = \operatorname{tg} x = x + \frac{x^3}{3} + \frac{2}{15} x^5 + \frac{17}{315} x^7 + \frac{62}{2835} x^9 + \frac{125762}{14189175} x^{11} + \dots, |x| < \frac{\pi}{2}, \varepsilon = 5 \cdot 10^{-4}.$$

$$4. f(x) = \operatorname{ctg} x = \frac{1}{x} - \frac{x}{3} - \frac{x^3}{45} - \frac{2}{945} x^5 - \frac{1}{4725} x^7 - \frac{2}{93555} x^9 + \dots, |x| < \pi, \\ \varepsilon = 10^{-4}.$$

$$5. f(x) = \sec x = 1 + \frac{x^2}{2} + \frac{5}{24} x^4 + \frac{61}{720} x^6 + \frac{277}{8064} x^8 + \frac{50521}{3628800} x^{10} + \dots, |x| < \frac{\pi}{2}, \\ \varepsilon = 1.5 \cdot 10^{-3}.$$

$$6. f(x) = \operatorname{cosec} x = \frac{1}{x} + \frac{x}{6} + \frac{7 \cdot x^3}{360} + \frac{31}{15120} x^5 + \frac{127}{604800} x^7 + \frac{511}{23950080} x^9 + \dots, |x| < \pi, \\ \varepsilon = 10^{-4}.$$

$$7. f(x) = \operatorname{sh} x = x + \frac{x^3}{3!} + \frac{x^5}{5!} + \dots + \sum_{k=0}^{\infty} \frac{x^{2k+1}}{(2k+1)!} + \dots, |x| < \infty, \varepsilon = 10^{-3}.$$

$$8. f(x) = \operatorname{ch} x = 1 + \frac{x^2}{2!} + \frac{x^4}{4!} + \dots + \sum_{k=0}^{\infty} \frac{x^{2k}}{(2k)!} + \dots, |x| < \infty, \varepsilon = 10^{-3}.$$

$$9. f(x) = \operatorname{th} x = x - \frac{x^3}{3} + \frac{2}{15} x^5 - \frac{17}{315} x^7 + \frac{62}{2835} x^9 - \frac{125762}{14189175} x^{11} + \dots, |x| < \frac{\pi}{2}, \\ \varepsilon = 5 \cdot 10^{-4}.$$

$$10. f(x) = \operatorname{cth} x = \frac{1}{x} + \frac{x}{3} - \frac{x^3}{45} + \frac{2}{945} x^5 - \frac{1}{4725} x^7 + \frac{2}{93555} x^9 + \dots, |x| < \pi, \\ \varepsilon = 10^{-4}.$$

$$11. f(x) = \sec h x = \frac{1}{\operatorname{ch} x} = 1 - \frac{x^2}{2!} + \frac{5}{4!} x^4 - \frac{61}{6!} x^6 + \frac{1385}{8!} x^8 - \frac{50521}{10!} x^{10} + \dots, \\ |x| < \frac{\pi}{2}, \varepsilon = 1.5 \cdot 10^{-3}.$$

$$12. f(x) = \operatorname{csch} x = \frac{1}{\operatorname{sh} x} = \frac{1}{x} - \frac{x}{6} + \frac{7 \cdot x^3}{360} - \frac{31}{15120} x^5 + \frac{127}{604800} x^7 - \frac{511}{23950080} x^9 + \dots, \\ |x| < \pi, \varepsilon = 10^{-4}.$$

$$13. f(x) = \frac{1}{2\sqrt{2}} \operatorname{arctg} \frac{x\sqrt{2}}{1-x^2} + \frac{1}{4\sqrt{2}} \ln \frac{1+x\sqrt{2}+x^2}{1-x\sqrt{2}+x^2} = \sum_{k=0}^{\infty} \frac{(-1)^k}{4k+1} x^{4k+1}, \\ |x| \leq 1, \varepsilon = 10^{-3}.$$

$$14. f(x) = \frac{1}{3} \operatorname{arctg} x + \frac{1}{6} \operatorname{arctg} \frac{x}{1-x^2} + \frac{1}{12\sqrt{3}} \ln \frac{1+x\sqrt{3}+x^2}{1-x\sqrt{3}+x^2} = \sum_{k=0}^{\infty} \frac{(-1)^k}{6k+1} x^{6k+1}, \\ |x| \leq 1, \varepsilon = 1.5 \cdot 10^{-3}.$$

$$15. f(x) = x + (1-x) \ln(1-x) = \sum_{k=1}^{\infty} \frac{x^{k+1}}{k(k+1)}, |x| \leq 1, \varepsilon = 10^{-3}.$$

$$16. f(x) = \frac{x}{2} + \frac{x^2}{4} + \frac{1}{2}(1-x^2) \ln(1-x) = \sum_{k=1}^{\infty} \frac{x^{k+2}}{k(k+2)}, |x| \leq 1, \varepsilon = 10^{-3}.$$

$$17. f(x) = \frac{3x^2}{4} - \frac{x}{2} - \frac{1}{2}(1-x^2) \ln(1-x) = \sum_{k=1}^{\infty} \frac{x^{k+2}}{k(k+1)(k+2)}, |x| \leq 1, \varepsilon = 5 \cdot 10^{-4}.$$

$$18. f(x) = \frac{1}{36} [11x^3 - 15x^2 + 6x + 6(1-x)^3 \ln(1-x)] = \sum_{k=1}^{\infty} \frac{x^{k+3}}{k(k+1)(k+2)(k+3)}, \\ |x| \leq 1, \varepsilon = 10^{-4}.$$

$$19. f(x) = \frac{4}{4-x^2} + 4x(4-x^2)^{-\frac{3}{2}} \arcsin \frac{x}{2} = \sum_{k=0}^{\infty} \frac{(k!)^2}{(2k)!} x^{2k}, |x| < 2, \varepsilon = 10^{-3}.$$

$$20. f(x) = 4(4-x^2)^{-\frac{1}{2}} \arcsin \frac{x}{2} = \sum_{k=0}^{\infty} \frac{(k!)^2}{(2k+1)!} x^{2k+1}, |x| < 2, \varepsilon = 10^{-3}.$$

$$21. f(x) = 2 \left(\arcsin \frac{x}{2} \right)^2 = \sum_{k=0}^{\infty} \frac{(k!)^2}{(2k+2)!} x^{2k+2}, |x| \leq 2, \varepsilon = 10^{-3}.$$

$$22. f(x) = 2x - 2(4-x^2)^{\frac{1}{2}} \arcsin \frac{x}{2} = \sum_{k=0}^{\infty} \frac{k!(k+1)!}{(2k+3)!} x^{2k+3}, |x| \leq 2, \varepsilon = 2 \cdot 10^{-3}.$$

$$23. f(x) = \ln(1+x) = x - \frac{x^2}{2} + \frac{x^3}{3} - \frac{x^4}{4} + \dots + \sum_{k=1}^{\infty} (-1)^{k+1} \frac{x^k}{k} + \dots, -1 < x \leq 1, \varepsilon = 3 \cdot 10^{-3}.$$

$$24. f(x) = \ln \left(\frac{x}{2} + \sqrt{\frac{x^2}{4} + 1} \right) = \frac{x}{2} - \frac{1}{2 \cdot 3} \cdot \frac{x^3}{2^3} + \frac{1 \cdot 3}{2 \cdot 4 \cdot 5} \cdot \frac{x^5}{2^5} - \frac{1 \cdot 3 \cdot 5}{2 \cdot 4 \cdot 6 \cdot 7} \cdot \frac{x^7}{2^7} + \dots, \\ |x| < 2, \varepsilon = 10^{-3}.$$

$$25. f(x) = \ln \cos x = -\frac{x^2}{2} - \frac{x^4}{12} - \frac{x^6}{45} - \frac{17x^8}{2520} - \frac{511x^{10}}{467775} + \dots, |x| < \frac{\pi}{2}, \varepsilon = 10^{-3}.$$

$$26. f(x) = \frac{1}{18} [6x + 3x^2 + 2x^3 + 6(1-x^3) \ln(1-x)] = \sum_{k=1}^{\infty} \frac{x^{k+3}}{k(k+3)}, |x| \leq 1, \varepsilon = 10^{-3}.$$

$$27. f(x) = -2x \operatorname{arctg} x + \ln(1+x^2) = \sum_{k=0}^{\infty} \frac{(-1)^k x^{2k+2}}{(k+1)(2k+1)}, |x| \leq 1, \varepsilon = 2 \cdot 10^{-3}.$$

$$28. f(x) = \frac{x}{2} + \frac{1}{4\sqrt{2}} (x^2 - 1) \operatorname{arctg} \frac{x\sqrt{2}}{1-x^2} - \frac{1}{8\sqrt{2}} (x^2 + 1) \ln \frac{1+x\sqrt{2}+x^2}{1-x\sqrt{2}+x^2} = \\ = \sum_{k=0}^{\infty} \frac{(-1)^k x^{4k+5}}{(4k+3)(4k+5)}, |x| \leq 1, \varepsilon = 10^{-3}.$$

$$29. f(x) = \frac{1}{16\sqrt{2}} (x^4 - 1) \left(2 \operatorname{arctg} \frac{x\sqrt{2}}{1-x^2} - \ln \frac{1+x\sqrt{2}+x^2}{1-x\sqrt{2}+x^2} \right) = \sum_{k=0}^{\infty} \frac{(-1)^k x^{4k+7}}{(4k+3)(4k+7)}, \\ |x| \leq 1, \varepsilon = 10^{-3}.$$

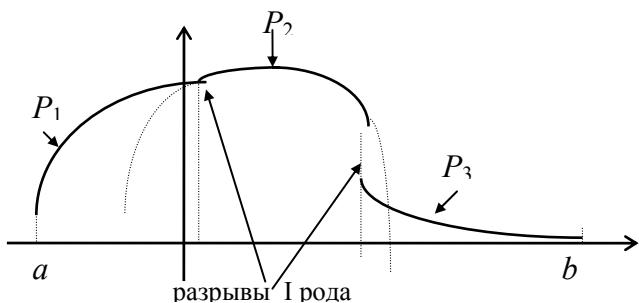
$$30. f(x) = \frac{1}{2} (\operatorname{sh} x - \sin x) = \sum_{k=0}^{\infty} \frac{x^{4k+3}}{(4k+3)!}, |x| < \infty, \varepsilon = 10^{-4}.$$

3.9. Локальная интерполяция

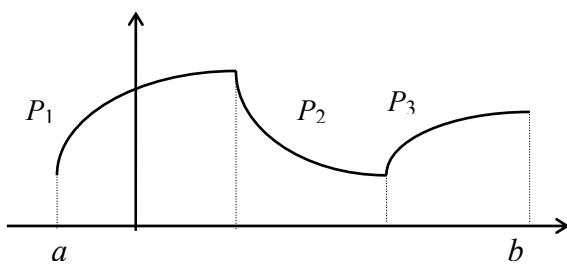
Как было показано ранее, глобальная полиномиальная интерполяция не всегда приводит к желательному заявленному результату по ряду причин. Альтернативный подход состоит в локальной интерполяции, когда функция $y = f(x)$ аппроксимируется интерполяционным многочленом $P_m(x)$ малой степени m на $[\alpha, \beta] \subset [a, b]$. При этом, конечно, используется лишь часть табличных данных.

Самый простой подход при локальной интерполяции состоит в построении набора полиномов $P(0,1,\dots,m), P(1,2,\dots,m+1), \dots, P(n-m,n-m+1,\dots,n)$ фиксированной степени m ,

каждый из которых совпадает с табличными значениями в $m+1$ последовательной точке. Каждый такой полином используется для интерполяции тех $x \in [a, b]$, для которых выбранные узлы таблицы являются ближайшими. Ясно, что в этом случае имеются разрывы I рода интерполированной функции в местах стыка многочленов



P_i . Этот метод называется интерполяцией с помощью «движущегося» полинома.



Другой подход состоит в кусочно-полиномиальной интерполяции. В этом случае исходный отрезок $[a, b]$ разбивается на несколько отрезков меньшей длины, на каждом из которых $y = f(x)$ интерполируется своим многочленом. При этом аппроксимируемая функция будет непрерывной, но будет иметь точки излома, то есть не будет непрерывно дифференцируемой.

3.10. Сплайны, их свойства и построение

В 1946 году были созданы сплайн-функции, являющиеся по сути дальнейшим развитием метода кусочно-полиномиальной интерполяции. Пусть отрезок $[a, b]$ разбит точками $x_0 = a < x_1 < \dots < x_{n-1} < x_n = b$ на n частичных отрезков $[x_{i-1}, x_i]$.

Сплайном степени m называется функция $S_m(x)$, удовлетворяющая следующим свойствам:

- 1) $S_m(x)$ непрерывна на $[a, b]$ вместе со всеми производными $S_m^{(1)}(x), S_m^{(2)}(x), \dots, S_m^{(p)}(x)$ до некоторого порядка p ;
- 2) на каждом отрезке $[x_{i-1}, x_i] \subset [a, b]$ $S_m(x)$ совпадает с некоторым алгебраическим многочленом $P_{m,i}(x)$ степени m .

Разность $m - p$ между степенью сплайна и порядком старшей производной, непрерывной на $[a, b]$, называется дефектом сплайна.

Интерполяция ломаными-простейший случай сплайна-сплайна первой степени дефекта единицы, где сама функция непрерывна, а уже первая производная разрывна. Наиболее широкое применение на практике получили сплайны $S_3(x)$ - кубические сплайны с дефектом, равным единице или двойке. Такие сплайны на каждом из частичных отрезков $[x_{i-1}, x_i]$ совпадают с кубическим многочленом $S_3(x) = P_{3,i}(x) = a_i + b_i(x - x_{i-1}) + c_i(x - x_{i-1})^2 + d_i(x - x_{i-1})^3$ и имеют на отрезке $[a, b]$ по крайней мере одну непрерывную производную $S'_3(x)$.

a). Интерполяционный сплайн.

Пусть на концах отрезка $[x_0, x_1]$ заданы значения y_0, y_1, y'_0 и y'_1 . Тогда интерполяционный многочлен $P_3(x)$, удовлетворяющий условиям $P_3(x_0) = y_0, P_3(x_1) = y_1, P'_3(x_0) = y'_0, P'_3(x_1) = y'_1$, может быть представлен в следующем виде:

$$P_3(x) = y_0 \frac{(x_1 - x)^2(2(x - x_0) + h)}{h^3} + y'_0 \frac{(x_1 - x)^2(x - x_0)}{h^2} + \\ + y_1 \frac{(x - x_0)^2(2(x_1 - x) + h)}{h^3} + y'_1 \frac{(x - x_0)^2(x - x_1)}{h^2}, \quad h = x_1 - x_0. \quad (3.10.1)$$

Эта формула проверяется непосредственно подстановкой граничных условий. Многочлен (3.10.1) называется кубическим интерполяционным многочленом Эрмита*.

Сплайн $S_m(x)$ называется интерполяционным, если $S_m(x_i) = y_i$ для всех $i = \overline{0, n}$. Если определять этот сплайн на $[x_{i-1}, x_i]$, то необходимо задать $y_{i-1}, y_i, s_{i-1}, s_i$, где $s_i = S'_m(x_i)$. Тогда по формуле (3.10.1) $S_3(x) = P_{3,i}(x)$.

* Шарль Эрмит (1822-1901) - французский математик.

б). Глобальный сплайн.

Для того, чтобы на всем отрезке $[a, b]$ сплайн имел непрерывную p -ю производную, в узлах сопряжения необходимо обеспечить условия касания p -го порядка, то есть

$$\begin{aligned} f(x_i) &= P_m(x_i), \\ f'(x_i) &= P'_m(x_i), \end{aligned}$$

$$\dots$$

$$f^{(p)}(x_i) = P_m^{(p)}(x_i).$$

Для кубического сплайна необходимо позаботиться о первой и второй производной в узлах сопряжения.

Сплайн-функции математически моделируют очень старое механическое устройство: гибкие рейки представляют собой механический сплайн. Если их жестко закрепить в узлах интерполяции, то рейки принимают форму, минимизирующую их потенциальную энергию, пропорциональную интегралу по длине дуги от квадрата кривизны рейки. Итак, если $S(x)$ -сплайн-функция, то $\int_{x_1}^{x_n} [S''(x)]^2 dx \Rightarrow \min$. С математической точки зрения доказано, что

сплайн - это единственная функция, обладающая свойством минимальной кривизны среди всех функций, интерполирующих данные точки, то есть это самая гладкая из функций, применяемых для данной интерполяции. Из определения сплайна и определения касания p -го порядка ясно, что если $x_i - i$ -й узел, то

$$\begin{aligned} S_m(x_i - 0) &= S_m(x_i + 0), \\ S'_m(x_i - 0) &= S'_m(x_i + 0), \end{aligned}$$

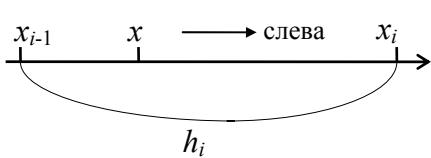
$$\dots$$

$$S_m^{(p)}(x_i - 0) = S_m^{(p)}(x_i + 0).$$

Рассмотрим весь отрезок $[a, b]$. Построим на нем систему сплайнов $S_3(x)$ с узлами сопряжения $a = x_0 < x_1 < x_2 < \dots < x_{n-1} < x_n = b$ с непрерывной второй производной. Для этого необходимо выбирать наклоны $s_i = S'_3(x)$ так, чтобы в точках x_i «стыка» многочленов $P_{3,i}$ и $P_{3,i+1}$ совпали значения их вторых производных:

$$P''_{3,i}(x_i) = P''_{3,i+1}(x_i), \quad i = 1, 2, \dots, n-1.$$

Продифференцируем формулу (3.10.1) и найдем вторую производную, переписав эту формулу для узлов x_{i-1} и x_i и вводя обозначения



$$s_i = S'_{3,i}(x_i);$$

$$S_3(x) = P_{3,i}(x) = \frac{(x - x_i)^2(2(x - x_{i-1}) + h_i)}{h_i^3} y_{i-1} +$$

$$+ \frac{(x - x_{i-1})^2(2(x_i - x) + h_i)}{h_i^3} y_i + \frac{(x - x_i)^2(x - x_{i-1})}{h_i^2} s_{i-1} +$$

$$+ \frac{(x - x_{i-1})^2(x - x_i)}{h_i^2} s_i, \quad h_i = x_i - x_{i-1}. \quad \text{Отсюда } S'_3(x) = P'_{3,i}(x) =$$

$$= \frac{2(x - x_i)(2(x - x_{i-1}) + h_i) + 2(x - x_i)^2}{h_i^3} y_{i-1} + \frac{2(x - x_{i-1})(2(x_i - x) + h_i) - 2(x - x_{i-1})^2}{h_i^3} y_i +$$

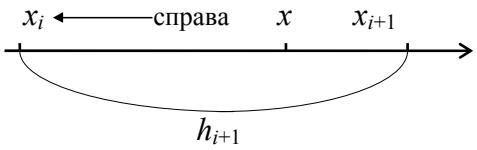
$$+ \frac{2(x - x_i)(x - x_{i-1}) + (x - x_i)^2}{h_i^2} s_{i-1} + \frac{2(x - x_{i-1})(x - x_i) + (x - x_{i-1})^2}{h_i^2} s_i. \quad \text{Аналогично}$$

$$S_3''(x) = P_{3,i}''(x) = \frac{2(2(x - x_{i-1}) + h_i) + 8(x - x_i)}{h_i^3} y_{i-1} + \frac{2(2(x_i - x) + h_i) - 8(x - x_{i-1})}{h_i^3} y_i +$$

$$+ \frac{2(x - x_{i-1}) + 4(x - x_i)}{h_i^2} s_{i-1} + \frac{2(x - x_i) + 4(x - x_{i-1})}{h_i^2} s_i. \quad \text{Положим } x = x_i, \quad \text{получим}$$

$$S_3''(x_i - 0) = P_{3,i}''(x_i) = \frac{4(x_i - x_{i-1}) + 2h_i}{h_i^3} y_{i-1} + \frac{2h_i - 8(x_i - x_{i-1})}{h_i^3} y_i +$$

$$+ \frac{2(x_i - x_{i-1})}{h_i^2} s_{i-1} + \frac{4(x_i - x_{i-1})}{h_i^2} s_i = \frac{2s_{i-1}}{h_i} + \frac{4s_i}{h_i} - 6 \frac{y_i - y_{i-1}}{h_i^2}.$$

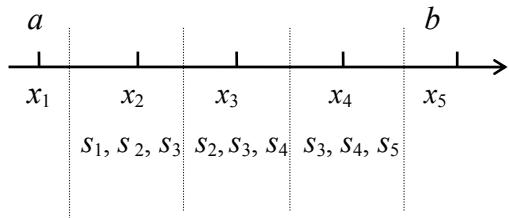


Аналогично получается вторая производная справа. Перепишем $P_{3,i}''(x)$ для узлов x_i, x_{i+1} , заменив x_i на x_{i+1} и x_{i-1} на x_i . Полагая затем $x = x_i$, в полученной формуле будем иметь

$$S_3''(x_i + 0) = P_{3,i+1}''(x_i) = -\frac{4s_i}{h_{i+1}} - \frac{2s_{i+1}}{h_{i+1}} + 6 \frac{y_{i+1} - y_i}{h_{i+1}^2}. \quad \text{Таким образом, записывая условия сопряжения второй производной во всех узлах промежутка } [a, b], \text{ получим следующую систему уравнений:}$$

$$h_i^{-1} s_{i-1} + 2(h_i^{-1} + h_{i+1}^{-1}) s_i + h_{i+1}^{-1} s_{i+1} = 3[h_i^{-2}(y_i - y_{i-1}) + h_{i+1}^{-2}(y_{i+1} - y_i)], \quad i = 1, 2, \dots, n-1. \quad (3.10.2)$$

Неизвестными в этой системе служат s_{i-1}, s_i, s_{i+1} . Число неизвестных $n+1$, а число



уравнений $n-1$. Таким образом, не хватает еще двух условий. Их выбор обычно связывают с некоторыми дополнительными условиями, накладываемыми на сплайн в граничных точках a и b (граничные условия). Эти условия могут быть, например, таковы.

1. Если в граничных точках известны значения первой производной $f'(a)$ и $f'(b)$, то естественно положить $s_0 = f'(a)$, $s_n = f'(b)$. (3.10.3)

Дополняя систему (3.10.2) уравнениями (3.10.3), приходим к системе уравнений с трехдиагональной матрицей, которая легко решается методом прогонки. Полученный таким образом сплайн называется фундаментальным кубическим сплайном.

2. Если в граничных точках известны $f''(a)$ и $f''(b)$, то $S_3''(a) = P_{3,1}''(x_0) = f''(a)$ и $S_3''(b) = P_{3,n}''(x_n) = f''(b)$. Это дает следующие уравнения:

$$\text{а) слева } P_{3,i}''(x_i) = \frac{2s_{i-1}}{h_i} + \frac{4s_i}{h_i} - 6 \frac{y_i - y_{i-1}}{h_i^2}, \quad i = n, \text{ тогда } P_{3,n}''(x_n) = f''(b) = \frac{2s_{n-1}}{h_n} +$$

$$+ \frac{4s_n}{h_n} - 6 \frac{y_n - y_{n-1}}{h_n^2}, \quad \text{б) справа } P_{3,i+1}''(x_i) =$$

$$= -\frac{4s_i}{h_{i+1}} - \frac{2s_{i+1}}{h_{i+1}} + 6 \frac{y_{i+1} - y_i}{h_{i+1}^2}, \quad i = 0, \quad \text{тогда}$$

$$P_{3,1}''(x_0) = f''(a) = -\frac{4s_0}{h_1} - \frac{2s_1}{h_1} + 6 \frac{y_1 - y_0}{h_1^2}.$$

3. Можно положить $f''(a) = 0$, $f''(b) = 0$, тогда соответствующая система уравнений определяет так называемый естественный кубический сплайн. Благодаря большой простоте записи они получили широкое распространение, однако условие $f''(a) = 0$, $f''(b) = 0$ приводит к потере точности и, таким образом, естественный сплайн обладает лишь вторым поряд-

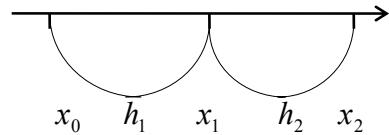
ком точности, в то время как все кубические сплайны имеют четвертый порядок точности относительно шага сетки узлов сопряжения, то есть $O(h^4)$.

4. Если нет никакой информации о значениях производных на концах отрезка, то прибегают к условию «отсутствие узла». Используются равенства $P_{3,1}(x) \equiv P_{3,2}(x)$ и $P_{3,n-1}(x) \equiv P_{3,n}(x)$, для чего необходимо, чтобы $P_{3,1}^{(3)}(x_1) = P_{3,2}^{(3)}(x_1)$ $P_{3,n-1}^{(3)}(x_{n-1}) = P_{3,n}^{(3)}(x_{n-1})$.

$$\text{Выведем необходимые формулы: } P_{3,i}^{(3)}(x) = \frac{4+8}{h_i^3} y_{i-1} + \frac{-4-8}{h_i^3} y_i + \frac{6}{h_i^2} s_{i-1} + \frac{6}{h_i^2} s_i =$$

$$= \frac{12}{h_i^3} y_{i-1} - \frac{12}{h_i^3} y_i + \frac{6}{h_i^2} s_{i-1} + \frac{6}{h_i^2} s_i.$$

a). Слева ($x = x_1$) $P_{3,1}^{(3)}(x_1) = \frac{12}{h_1^3} y_0 - \frac{12}{h_1^3} y_1 + \frac{6}{h_1^2} s_0 + \frac{6}{h_1^2} s_1.$



б). Справа ($x = x_1$) $P_{3,2}^{(3)}(x_1) = \frac{12}{h_2^3} y_1 - \frac{12}{h_2^3} y_2 + \frac{6}{h_2^2} s_1 + \frac{6}{h_2^2} s_2.$ Так как $P_{3,1}^{(3)}(x_1) = P_{3,2}^{(3)}(x_1)$, то

$$\begin{aligned} & \frac{2y_0}{h_1^3} - \frac{2y_1}{h_1^3} + \frac{s_0}{h_1^2} + \frac{s_1}{h_1^2} = \frac{2y_1}{h_2^3} - \frac{2y_2}{h_2^3} + \frac{s_1}{h_2^2} + \frac{s_2}{h_2^2}. \quad \text{Отсюда получим } 2h_1^{-3}(y_0 - y_1) + h_1^{-2}(s_0 + s_1) = \\ & = 2h_2^{-3}(y_1 - y_2) + h_2^{-2}(s_1 + s_2). \end{aligned}$$

Такие же преобразования для узлов x_{n-2}, x_{n-1}, x_n дают аналогичное уравнение

$$2h_{n-1}^{-3}(y_{n-2} - y_{n-1}) + h_{n-1}^{-2}(s_{n-2} + s_{n-1}) = 2h_n^{-3}(y_{n-1} - y_n) + h_n^{-2}(s_{n-1} + s_n).$$

Можно доказать, что условие 4 эквивалентно объединению отрезков $[x_0, x_1] + [x_1, x_2]$ и $[x_{n-2}, x_{n-1}] + [x_{n-1}, x_n]$ и построению сплайнов по системе $n-2$ отрезков внутри $[a, b]$. Этот факт объясняет название «отсутствие узла».

5. Если $y = f(x)$ - периодическая функция с периодом $b-a$, то можно воспользоваться условием $s_0 = s_n$. Тогда $h_n^{-1}(s_{n-1} + 2s_n) + h_1^{-1}(2s_0 + s_1) = 3[h_n^{-2}(y_n - y_{n-1}) + h_1^{-2}(y_1 - y_0)]$ (выводится аналогично из условия $P_{3,0}^{(3)}(x_0) = P_{3,n}^{(3)}(x_n)$). Существуют и другие подходы к заданию граничных условий.

Запишем теперь, например, с условием 4 всю систему, симметризовав предварительно первое и последнее уравнение. Получим

$$\begin{cases} h_1^{-2}s_0 + (h_1^{-2} - h_2^{-2})s_1 - h_2^{-2}s_2 = 2h_2^{-3}(y_1 - y_2) - 2h_1^{-3}(y_0 - y_1), & i = 0, \\ h_i^{-1}s_{i-1} + 2(h_i^{-1} + h_{i+1}^{-1})s_i + h_{i+1}^{-1}s_{i+1} = 3[h_i^{-2}(y_i - y_{i-1}) + h_{i+1}^{-2}(y_{i+1} - y_i)], & i = 1, 2, \dots, n-1, \\ h_{n-1}^{-2}s_{n-2} + (h_{n-1}^{-2} - h_n^{-2})s_{n-1} - h_n^{-2}s_n = 2h_n^{-3}(y_{n-1} - y_n) - 2h_{n-1}^{-3}(y_{n-2} - y_{n-1}), & i = n. \end{cases} \quad (3.10.4)$$

Эта система, записанная в матричном виде, приводит к трехдиагональной матрице. Действительно,

$$\begin{pmatrix} h_1^{-2} & h_1^{-2} + h_2^{-2} & -h_2^{-2} & 0 & 0 & 0 & 0 \\ h_1^{-1} & 2(h_1^{-1} + h_2^{-1}) & h_2^{-1} & 0 & 0 & 0 & 0 \\ 0 & h_2^{-1} & 2(h_2^{-1} + h_3^{-1}) & h_3^{-1} & 0 & 0 & 0 \\ \dots & \dots & \dots & \dots & \dots & \dots & \dots \\ 0 & 0 & 0 & h_{n-2}^{-1} & 2(h_{n-2}^{-1} + h_{n-1}^{-1}) & h_{n-1}^{-1} & 0 \\ 0 & 0 & 0 & 0 & h_{n-1}^{-1} & 2(h_{n-1}^{-1} + h_n^{-1}) & h_n^{-1} \\ 0 & 0 & 0 & 0 & h_{n-1}^{-2} & h_{n-1}^{-2} + h_n^{-2} & -h_n^{-2} \end{pmatrix} \cdot \begin{pmatrix} s_0 \\ s_1 \\ s_2 \\ \dots \\ s_{n-2} \\ s_{n-1} \\ s_n \end{pmatrix} = \begin{pmatrix} b_0 \\ b_1 \\ b_2 \\ \dots \\ b_{n-2} \\ b_{n-1} \\ b_n \end{pmatrix},$$

где $b_0 = 2h_2^{-3}(y_1 - y_2) - 2h_1^{-3}(y_0 - y_1)$, $b_1 = 3[h_1^{-2}(y_1 - y_0) + h_2^{-2}(y_2 - y_1)]$...,

$b_n = 2h_n^{-3}(y_{n-1} - y_n) - 2h_{n-1}^{-3}(y_{n-2} - y_{n-1})$. Такие системы быстро решаются специальными методами, с которыми мы познакомимся позднее.

3.11. Погрешность приближения кубическими сплайнами

Теорема 3.10. Пусть функция $y = f(x)$ имеет на отрезке $[a, b]$ непрерывную производную четвертого порядка и $M_4 = \max_{[a,b]} |f^{(4)}(x)|$. Тогда для интерполяционного кубического сплайна $S_3(x)$, удовлетворяющего граничным условиям типов 1, 2, 4 и 5, справедлива следующая оценка погрешности: $\max_{[a,b]} |f(x) - S_3(x)| \leq C \cdot M_4 h_{\max}^4$, где C - некоторая константа.

Характерно, что сплайн $S_3(x)$ не только сам аппроксимирует функцию $y = f(x)$, но и его производные $S'_3(x), S''_3(x), S'''_3(x)$ приближают соответствующие производные функции $y = f(x)$.

Теорема 3.11. При выполнении условий теоремы 3.10 для указанных в ней сплайнов справедливы неравенства $\max_{[a,b]} |f^{(k)}(x) - S_3^{(k)}(x)| \leq C_k M_4 h^{4-k}$, $k = 1, 2, 3$.

Пример. Пусть функция задана следующей таблицей:

i	0	1	2	3	4
x_i	0	1	2	3	4
y_i	1.0	1.8	2.2	1.4	1.0

Построить для этой функции сплайн с граничными условиями 4 на концах отрезка. Здесь $n = 4$, $h = 1 = \text{const}$. Запишем систему (3.10.4), реализующую эту схему в конкретном числовом выражении. Первое уравнение имеет вид $h_1^{-2}s_0 + (h_1^{-2} - h_2^{-2})s_1 - h_2^{-2}s_2 = 2h_2^{-3}(y_1 - y_2) - 2h_1^{-3}(y_0 - y_1)$. Вычисляя значения $h_i - h_{i-1}$ и подставляя $h = 1$, получим $s_0 + 0 \cdot s_1 - s_2 = 2 \cdot (-0.4) - 2 \cdot (-0.8) \Rightarrow s_0 - s_2 = 0.8$. Аналогично, последнее уравнение дает $h_{n-1}^{-2}s_{n-2} + (h_{n-1}^{-2} - h_n^{-2})s_{n-1} - h_n^{-2}s_n = 2h_n^{-3}(y_{n-1} - y_n) - 2h_{n-1}^{-3}(y_{n-2} - y_{n-1})$, $n = 4$, $n-1 = 3$, $n-2 = 3$. Тогда $h_3^{-2}s_2 + (h_3^{-2} - h_4^{-2})s_3 - h_4^{-2}s_4 = 2h_4^{-3}(y_3 - y_4) - 2h_3^{-3}(y_2 - y_3) \Rightarrow s_2 - 0 \cdot s_3 - s_4 = 2 \cdot 0.4 - 2 \cdot 0.8 \Rightarrow s_2 - s_4 = -0.8$. Три внутренние узла дадут три следующих уравнения:

$$\begin{aligned} i = 1 : h_1^{-1}s_0 + 2(h_1^{-1} + h_2^{-1})s_1 + h_2^{-1}s_2 &= 3[h_1^{-2}(y_1 - y_0) + h_2^{-2}(y_2 - y_1)], \\ i = 2 : h_2^{-1}s_1 + 2(h_2^{-1} + h_3^{-1})s_2 + h_3^{-1}s_3 &= 3[h_2^{-2}(y_2 - y_1) + h_3^{-2}(y_3 - y_2)], \\ i = 3 : h_3^{-1}s_2 + 2(h_3^{-1} + h_4^{-1})s_3 + h_4^{-1}s_4 &= 3[h_3^{-2}(y_3 - y_2) + h_4^{-2}(y_4 - y_3)] \end{aligned}$$

Упрощая их аналогичным способом, что и выше, получим

$$\begin{aligned} s_0 + 2 \cdot 2 \cdot s_1 + s_2 &= 3(0.8 + 0.4), \\ s_1 + 2 \cdot 2 \cdot s_2 + s_3 &= 3(0.4 + (-0.8)), \\ s_2 + 2 \cdot 2 \cdot s_3 + s_4 &= 3(-0.8 + (-0.4)). \end{aligned}$$

Тогда вся система имеет вид

$$\left\{ \begin{array}{l} s_0 - s_2 = 0.8, \\ s_0 + 4s_1 + s_2 = 3.6, \\ s_1 + 4s_2 + s_3 = -1.2, \\ s_2 + 4s_3 + s_4 = -3.6, \\ s_2 - s_4 = -0.8. \end{array} \right.$$

Решим эту систему методом Гаусса:

$$\left\{ \begin{array}{l} s_0 - s_2 = 0.8, \\ 4s_1 + 2s_2 = 2.8, \\ s_1 + 4s_2 + s_3 = -1.2, \\ s_2 + 4s_3 + s_4 = -3.6, \\ s_2 - s_4 = -0.8. \end{array} \right. \Rightarrow \left\{ \begin{array}{l} s_0 - s_2 = 0.8, \\ s_1 + 4s_2 + s_3 = -1.2, \\ -14s_2 - 4s_3 = 7.6, \\ s_2 + 4s_3 + s_4 = -3.6, \\ s_2 - s_4 = -0.8. \end{array} \right.$$

$$\left\{ \begin{array}{l} s_0 - s_2 = 0.8, \\ s_1 + 4s_2 + s_3 = -1.2, \\ s_2 + 4s_3 + s_4 = -3.6, \\ -4s_3 - 2s_4 = 2.8, \\ 52s_3 + 14s_4 = -42.8. \end{array} \right. \Rightarrow \left\{ \begin{array}{l} s_0 - s_2 = 0.8, \\ s_1 + 4s_2 + s_3 = -1.2, \\ s_2 + 4s_3 + s_4 = -3.6, \\ s_3 + 0.5s_4 = -0.7, \\ -12s_4 = -6.4. \end{array} \right.$$

Отсюда $s_4 = 8/15$, $s_3 = -29/30$, $s_2 = -4/15$, $s_1 = 5/6$, $s_0 = 8/15$. Запись формулы (3.10.1) кубического интерполяционного многочлена Эрмита, задавая y_{i-1} , y_i , s_{i-1} , s_i однозначно определяет сплайн на отрезке $[x_{i-1}, x_i]$. Эта формула уже учитывает, что интерполированная функция $f(x)$ будет непрерывна и один раз дифференцируема везде на сетке x_0, x_1, \dots, x_n . Таким образом, на каждом из четырех отрезков $[0, 1], [1, 2], [2, 3], [3, 4]$ конкретный вид сплайна можно установить по формуле (3.10.1). Например, для $[0, 1]$ имеем $y_0 = 1.0$, $y_1 = 1.8$, $s_0 = 8/15$, $s_1 = 5/6$. Тогда

$$P_{3,1}(x) = y_0 \frac{(x_1 - x)^2(2(x - x_0) + h)}{h^3} + y'_0 \frac{(x_1 - x)^2(x - x_0)}{h^2} + \\ + y_1 \frac{(x - x_0)^2(2(x_1 - x) + h)}{h^3} + y'_1 \frac{(x - x_0)^2(x - x_1)}{h^2}, \quad h = x_1 - x_0.$$

$$P_{3,1}(x) = 1 \cdot \frac{(1-x)^2(2(x-0)+1)}{1} + \frac{8}{15} \frac{(1-x)^2(x-0)}{1} + 1.8 \frac{(x-0)^2(2(1-x)+1)}{1} + \\ + \frac{5}{6} \frac{(x-0)^2(x-1)}{1} = (x-1)^2(2x+1) + \frac{8}{15}(x-1)^2x + 1.8x^2(3-2x) + \frac{5}{6}x^2(x-1) = \\ = -\frac{7}{30}x^3 + \frac{1}{2}x^2 + \frac{8}{15}x + 1 = -0.233333x^3 + 0.500000x^2 + 0.533333x + 1.000000.$$

Многочлены $P_{3,i}$ на остальных отрезках рассчитываются аналогично. Именно:

$$x \in [1, 2], \quad x_1 = 1, \quad x_2 = 2, \quad y_1 = 1.8, \quad y_2 = 2.2, \quad s_1 = 0.833333, \quad s_2 = -0.266667,$$

$$P_{3,2}(x) = -0.233333x^3 + 0.500000x^2 + 0.533333x + 1.000000.$$

$$x \in [2, 3], \quad x_2 = 2, \quad x_3 = 3, \quad y_2 = 2.2, \quad y_3 = 1.4, \quad s_2 = -0.266667, \quad s_3 = -0.966667,$$

$$P_{3,3}(x) = 0.366667x^3 - 3.100000x^2 + 7.700000x - 3.800000.$$

$$x \in [3, 4], \quad x_3 = 3, \quad x_4 = 4, \quad y_3 = 1.4, \quad y_4 = 1.0, \quad s_3 = -0.966667, \quad s_4 = 0.533333,$$

$$P_{3,4}(x) = 0.366667x^3 - 3.100000x^2 + 7.700000x - 3.800000.$$

3.12. Лабораторная работа № 6. Аппроксимация функций кубическими сплайнами

Часто на практике, для того чтобы аппроксимировать функцию, вместо построения глобального интерполяционного многочлена на всем промежутке определения аргумента используют интерполяцию кусочными многочленами – сплайнами.

Гладкие сплайны обладают многими замечательными свойствами, которые обеспечили им успех в приложениях. Эти свойства описаны в подразд 3.10. Решим в пакете Mathcad задачу аппроксимации таблично заданной функции с помощью кубических сплайнов.

Среди одиннадцати функций интерполяции и экстраполяции в среде Mathcad имеются средства линейной интерполяции (функция **linterp(vx,vy,x)** – значение в точке x линейного интерполяционного многочлена векторов vx и vy) и интерполяции сплайнами, осуществляющей функцией **interp(K,X,Y,x)** по значениям коэффициентов K векторов линейного, параболического или кубического сплайнов. Вектор коэффициентов соответствующего сплайна вычисляется подпрограммами **lspline(vx,vy)**, **pspline(vx,vy)** и **cspline(vx,vy)**.

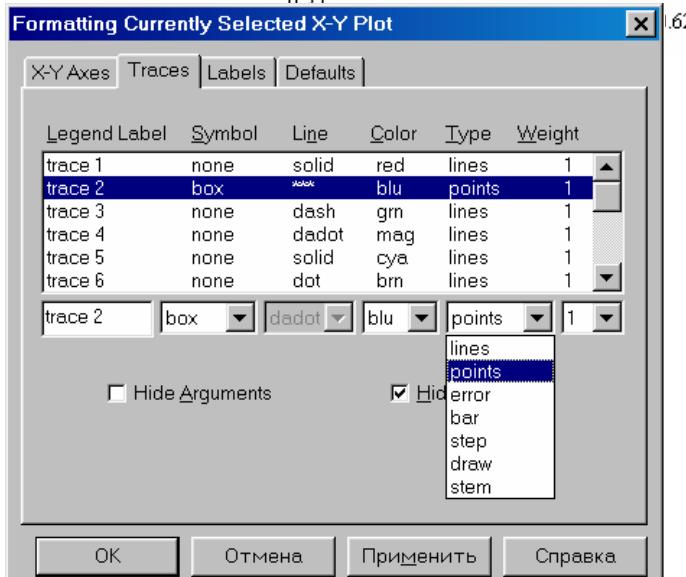
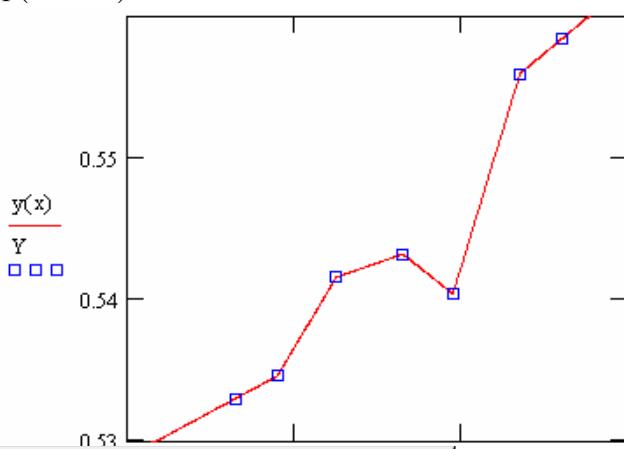
Различия в линейной, параболической и кубической интерполяции сплайнами заметно проявляются лишь на концах заданной сетки узлов.

Построим все три типа сплайнов и рассмотрим графическое представление данных с их помощью. Пусть

$$\text{ORIGIN} := 1 \quad X := \begin{pmatrix} 0.593 \\ 0.598 \\ 0.605 \\ 0.613 \\ 0.619 \\ 0.627 \\ 0.632 \end{pmatrix} \quad Y := \begin{pmatrix} 0.53305 \\ 0.53464 \\ 0.54160 \\ 0.54324 \\ 0.54043 \\ 0.55598 \\ 0.55843 \end{pmatrix}.$$

Проведем сначала линейную интерполяцию и построим график результата.

$$y(x) := \text{linterp}(X, Y, x)$$



Для того чтобы исходные данные представлялись в виде квадратиков, надо войти в панель форматирования графиков (Formatting Currently Selected X-Y Plot), щелкнув дважды левой кнопкой мыши по любому месту графика. В открывшейся панели

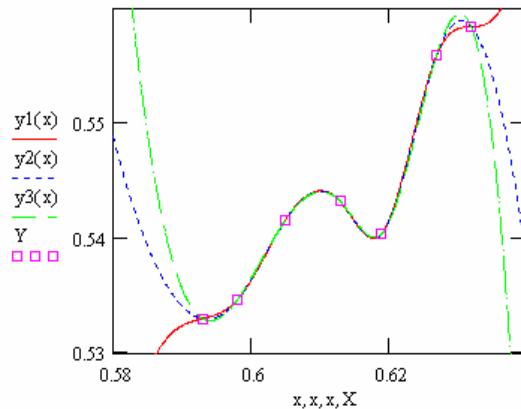
для графика № 2 (trace 2) следует выбрать представление кривой в виде точек (points) квадратной формы (box).

Сплайны соответствующей степени строятся аналогично. Наберем с клавиатуры

$$KLS := lspline(X, Y) \quad y1(x) := interp(KLS, X, Y, x)$$

$$KPS := pspline(X, Y) \quad y2(x) := interp(KPS, X, Y, x)$$

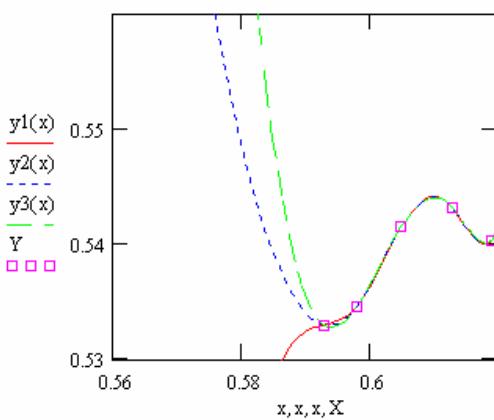
$$KCS := cspline(X, Y) \quad y3(x) := interp(KCS, X, Y, x).$$



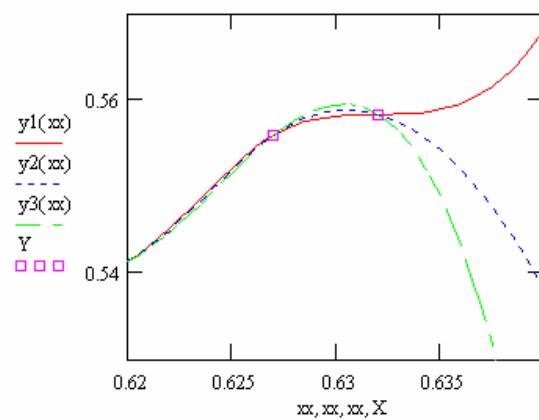
На всем отрезке $x \in [0.58, 0.64]$ графики имеют вид

На концах отрезка вблизи точек $x = 0.58$ и $x = 0.64$ все четыре графика построим отдельно:

$$x := 0.56, 0.5605 \dots 0.62$$



$$xx := 0.62, 0.6205 \dots 0.64$$



Явно видимое различие трех графиков проявляется лишь при экстраполяции; внутри отрезка $x \in [0.58, 0.64]$ все три графика почти сливаются в один.

Из всех четырех графиков видно, что функция **lspline**(X, Y) строит вектор коэффициентов линейного сплайна дефекта нуль, то есть во всех внутренних узлах сетки сстыкованы сама функция $y = y(x)$ и ее первая производная. Линейный сплайн дефекта один – это интерполяция ломаными, то есть функция **interp**(X, Y, x), график которой приведен на самом первом рисунке.

К сожалению, в доступной автору документации пакета Mathcad нет упоминания, какого типа краевые условия употреблялись для каждого вида сплайнов. Скорее всего, это были так называемые естественные условия, то есть условия равенства нулю старшей используемой производной на концах рассматриваемого отрезка.

Составим программу вычисления коэффициентов кубического сплайна с краевыми условиями четвертого типа – «отсутствие узла». Эти коэффициенты вычисляются решением системы линейных уравнений (3.10.4) методом прогонки. Система (3.10.4) отличается от классической трехдиагональной системы уравнений, используемой в методе прогонки, тем,

что в первом и последнем уравнении этой системы имеется по одному лишнему (ненулевому) коэффициенту. Поэтому формулы метода прогонки, приведенные в подразд. 5.4, необходимо несколько модернизировать. Модернизация касается способа вычисления x_1 и x_2 в прямом ходе и x_1 в обратном; все остальные переменные определяются стандартно по формулам (5.4.1) – (5.4.6).

Progonka1 – подпрограмма, реализующая модернизированный метод прогонки. Параметры: вектор b содержит коэффициенты при неизвестных левой части системы уравнений, стоящие на главной диагонали, вектор a – под главной диагональю, вектор c – над главной диагональю. Коэффициент p_1 первого уравнения системы $b_1x_1 + c_1x_2 + p_1x_3 = d_1$ находится в c_n , коэффициент p_2 последнего уравнения $p_2x_{n-2} + a_nx_{n-1} + b_nx_n = d_n$ в a_1 . Вектор d содержит свободные члены (правые части системы уравнений). Подпрограмма выдает вектор x – вектор решений системы. Для правильной работы подпрограммы необходимо задание $ORIGIN := 1$.

Следующая подпрограмма **intspline** вычисляет значение кубического сплайна с краевыми условиями четвертого типа для данного значения аргумента x . X и Y – векторы исходных данных, n – число точек сетки. Структура подпрограммы **intspline** совершенно прозрачна. Сначала вычисляются все коэффициенты диагональной системы (3.10.4), затем эта система решается модифицированным методом прогонки. Далее по значению аргумента x

```

progonka1(a,b,c,d) := | n ← rows(b)
                         | γ1 ← b1
                         | δ1 ← d1
                         | β1 ← δ1 / γ1
                         | ε1 ← -c1 / γ1
                         | δ1 ← -cn / b1
                         | γ2 ← b2 + a2 · ε1
                         | β2 ← (d2 - a2 · β1) / γ2
                         | ε2 ← -(c2 + a2 · δ1) / γ2
                         | for i ∈ 3..n - 1
                         |   γi ← bi + ai · εi-1
                         |   εi ← -ci / γi
                         |   βi ← (di - ai · βi-1) / γi
                         |   δi ← bn + εn-1 · (an + a1 · εn-2)
                         |   βn ← (dn - an · βn-1) / γn
                         |   εn ← -a1 · (εn-2 · βn-1 + βn-2) / γn
                         |   xn ← βn + εn
                         |   for i ∈ n - 1, n - 2..2
                         |     xi ← εi · xi+1 + βi
                         |     x1 ← β1 + ε1 · x2 + δ1 · x3
                         |   x

```

выбирается нужный кубический многочлен третьей степени $P_3(x)$ и находится его значение при данном x по формуле (3.10.1) кубического интерполяционного многочлена Эрмита.

В заключение приведем графики полученных кубических сплайнов в одном масштабе:

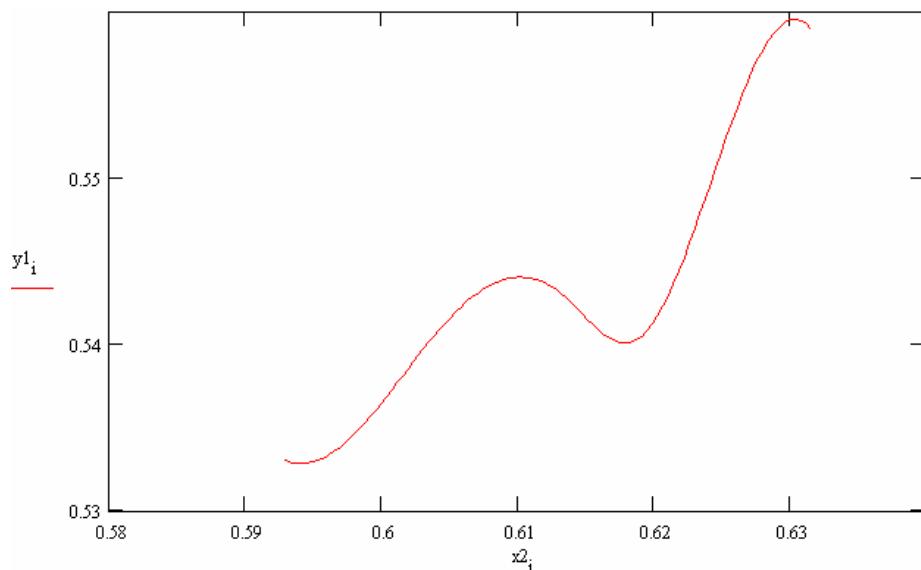
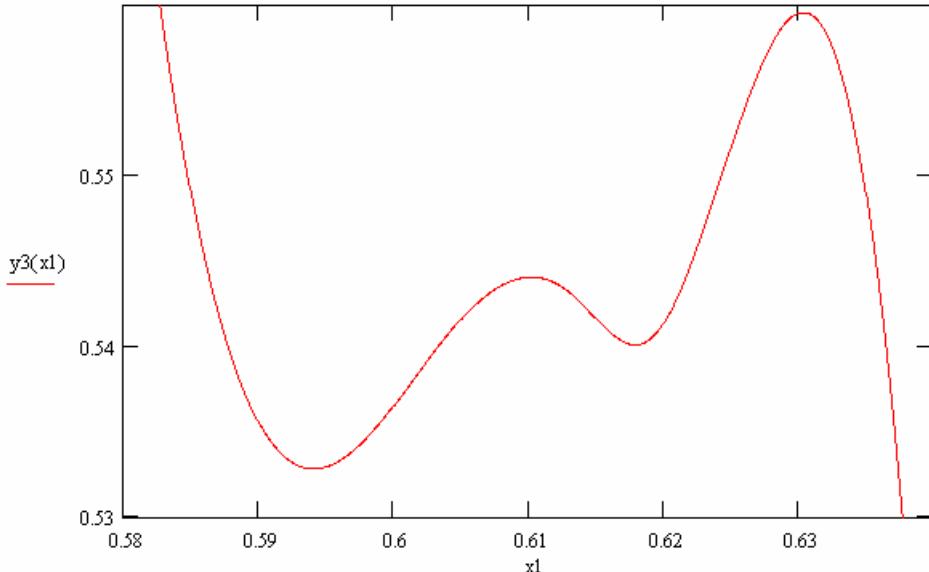
```

KCS := cspline(X,Y)  y3(x1):= interp(KCS,X,Y,x1)
i := 1...100  x2i := X1 + (X7 - X1) . (i - 1)
y1i := intspline(X,Y,7,x2i)
intspline(X,Y,n,x) := {
    h1 := X2 - X1
    h2 := X3 - X2
    b1 := 1 / h12
    cn := -1 / h22
    c1 := b1 + cn
    d1 := 2 * [ (cn / h2) * (Y3 - Y2) - b1 * (Y1 - Y2) ]
    h1 := Xn - Xn-1
    h2 := Xn-1 - Xn-2
    a1 := 1 / h22
    bn := -1 / h12
    an := a1 + bn
    dn := 2 * [ (bn / h1) * (Yn - Yn-1) - a1 / h2 * (Yn-2 - Yn-1) ]
    for i ∈ 2..n-1
        h1 := Xi - Xi-1
        h2 := Xi+1 - Xi
        ai := 1 / h1
        ci := 1 / h2
        bi := 2 * (ai + ci)
        di := 3 * [ (Yi - Yi-1) / h12 + (Yi+1 - Yi) / h22 ]
    xx := progonkal(a,b,c,d)
    k := 2
    for i ∈ 2..n
        if xx < Xi
            break
        k := i + 1
    h := Xk - Xk-1
    h1 := x - Xk-1
    h2 := Xk - x
}

```

$$\begin{aligned}
 p3 &\leftarrow \frac{Y_{k-1}}{h^3} \cdot h2^2 \cdot (2 \cdot h1 + h) + x_{k-1} \cdot h2^2 \cdot \frac{h1}{h^2} \\
 p3 &\leftarrow p3 + \frac{Y_k}{h^3} \cdot h1^2 \cdot (2 \cdot h2 + h) - x_k \cdot h1^2 \cdot \frac{h2}{h^2} \\
 | p3
 \end{aligned}$$

Видно, что оба графика на отрезке $x \in [0.59, 0.63]$ абсолютно совпадают.



В этом можно убедиться и сравнить полученные интерполированные значения $f(x)$ по двум разным программам:

$x5 := 0.6 \quad y5 := \text{interp}(KCS, X, Y, x5) \quad y5 = 0.536$

$y55 := \text{intspline}(X, Y, 7, x5) \quad y55 = 0.536$

$x6 := 0.63 \quad y6 := \text{interp}(KCS, X, Y, x6) \quad y6 = 0.560$

$y56 := \text{intspline}(X, Y, 7, x6) \quad y56 = 0.560.$

Задание № 1. Построить для таблично заданной функции $y = f(x)$ кубический сплайн с граничными условиями любого из четырех типов на концах отрезка:

Номера вариантов											
1		2		3		4		5			
x_i	y_i	x_i	y_i	x_i	y_i	x_i	y_i	x_i	y_i	x_i	y_i
0.698	2.2234	0.100	1.1213	0.235	1.2080	0.095	1.0913	0.103	2.0128		
0.706	2.2438	0.108	1.1316	0.240	1.2126	0.102	1.2349	0.108	2.0334		
0.714	2.2645	0.119	1.1459	0.250	1.2217	0.104	1.2799	0.115	2.0607		
0.727	2.2984	0.127	1.1565	0.255	1.2263	0.107	1.3514	0.120	2.0792		
0.736	2.3222	0.135	1.1671	0.265	1.2355	0.110	1.4282	0.128	2.1072		
0.747	2.3516	0.146	1.1819	0.280	1.2493	0.112	1.4826	0.136	2.1335		
0.760	2.3869	0.157	1.1969	0.295	1.2633	0.116	1.6003	0.141	2.1492		
0.769	2.4116	0.169	1.2134	0.300	1.2680	0.120	1.7321	0.150	2.1761		
0.782	2.4478	0.175	1.2196	0.305	1.2726	0.125	1.8500	0.157	2.1805		

Номера вариантов											
6		7		8		9		10			
x_i	y_i	x_i	y_i	x_i	y_i	x_i	y_i	x_i	y_i	x_i	y_i
0.296	3.2558	0.050	0.2079	0.902	1.2351	0.100	1.8378	0.400	1.6683		
0.303	3.1764	0.052	0.2081	0.909	1.2369	0.104	1.8369	0.405	1.6664		
0.310	3.1218	0.060	0.2090	0.919	1.2394	0.118	1.8335	0.410	1.6645		
0.317	3.0482	0.065	0.2095	0.940	1.2448	0.139	1.8286	0.420	1.6607		
0.323	2.9876	0.069	0.2099	0.944	1.2458	0.145	1.8272	0.429	1.6573		
0.330	2.9195	0.075	0.2105	0.955	1.2486	0.158	1.8242	0.440	1.6532		
0.339	2.8360	0.085	0.2116	0.965	1.2511	0.167	1.8221	0.449	1.6499		
0.345	2.7771	0.090	0.2121	0.975	1.2537	0.185	1.8179	0.455	1.6476		
0.352	2.6114	0.096	0.2127	1.010	1.2628	0.200	1.8145	0.465	1.6439		

Номера вариантов											
11		12		13		14		15			
x_i	y_i	x_i	y_i	x_i	y_i	x_i	y_i	x_i	y_i	x_i	y_i
1.030	2.8011	0.200	0.0400	0.010	1.0101	0.100	0.8100	1.000	0.0000		
1.080	2.9447	0.300	0.0899	0.105	1.1105	0.155	0.7140	1.510	0.2729		
1.160	3.1899	0.350	0.1222	0.156	1.1681	0.220	0.6084	2.100	0.3533		
1.230	3.4212	0.379	0.1431	0.200	1.2198	0.280	0.5184	2.750	0.3679		
1.260	3.5254	0.415	0.1714	0.215	1.2378	0.375	0.3906	3.420	0.3595		
1.330	3.7810	0.500	0.2474	0.289	1.3298	0.445	0.3080	3.915	0.3486		
1.390	4.0149	0.596	0.3478	0.316	1.3645	0.510	0.2401	4.350	0.3380		
1.450	4.1713	0.615	0.3693	0.390	1.4626	0.625	0.1406	4.800	0.3268		
1.500	4.2390	0.700	0.4706	0.500	1.6152	0.770	0.0529	5.200	0.3171		

Номера вариантов											
16		17		18		19		20			
x_i	y_i	x_i	y_i	x_i	y_i	x_i	y_i	x_i	y_i	x_i	y_i
0.100	0.0010	-2.100	-0.674	-1.000	1.368	1.000	0.000	1.000	0.6663		
0.210	0.0441	-1.815	0.914	-0.010	1.990	5.000	0.999	1.300	0.5706		
0.295	0.0868	-1.600	0.574	0.750	3.117	15.150	0.411	1.650	0.5429		
0.348	0.1205	-1.420	0.827	1.900	7.686	27.900	-0.186	1.915	0.5887		
0.419	0.1738	-1.290	0.931	2.815	17.692	35.100	-0.405	2.280	0.7256		
0.475	0.2219	-0.750	0.983	3.100	23.198	44.000	-0.599	2.500	0.8262		

0.511	0.2554	1.030	-0.497	3.419	31.539	59.500	-0.810	2.750	0.9280
0.555	0.2988	2.170	0.828	3.875	49.183	70.450	-0.897	2.935	0.9790
0.609	0.3552	2.500	-0.781	4.200	67.686	82.100	-0.954	3.150	0.9999
Номера вариантов									
21		22		23		24		25	
x_i	y_i	x_i	y_i	x_i	y_i	x_i	y_i	x_i	y_i
0.000	2.7183	0.100	-3.928	0.000	1.0000	0.500	0.9273	0.100	0.0905
0.425	2.4869	0.330	-3.743	0.175	0.9703	0.625	0.8938	0.300	0.2222
0.870	1.9057	0.519	-3.567	0.308	0.9133	0.800	0.8434	0.490	0.3002
1.395	1.1911	0.728	-3.344	0.514	0.7910	1.100	0.7573	0.730	0.3518
1.850	0.7591	0.997	-3.004	0.800	0.6098	1.450	0.6669	0.950	0.3674
2.410	0.4752	1.264	-2.598	0.996	0.5020	1.810	0.5890	1.200	0.3614
3.215	0.3689	1.513	-2.146	1.213	0.4046	2.250	0.5123	1.500	0.3347
3.397	0.3800	1.875	-1.332	1.456	0.3205	2.570	0.4668	1.710	0.3093
3.830	0.4620	2.011	-0.969	1.678	0.2621	3.000	0.4164	1.980	0.2734

Номера вариантов									
26		27		28		29		30	
x_i	y_i	x_i	y_i	x_i	y_i	x_i	y_i	x_i	y_i
1.100	0.438	1.500	0.919	1.000	2.3198	0.000	0.540	0.125	0.9058
1.250	0.709	3.750	0.247	1.500	2.7115	0.479	-0.044	0.180	0.9310
1.400	0.169	5.120	-0.062	1.985	2.4979	0.896	-0.770	0.236	0.9540
1.575	-0.042	8.300	-0.519	2.470	1.8631	1.365	-0.715	0.291	0.9730
1.720	-0.148	11.450	-0.723	2.990	1.1630	1.700	0.690	0.347	0.9879
1.900	-0.318	17.900	-0.967	3.500	0.7041	1.915	0.876	0.400	0.9969
2.070	-0.461	21.680	-0.998	3.875	0.5120	2.148	-0.655	0.462	0.9999
2.310	-0.624	25.100	-0.997	4.218	0.4147	2.583	0.784	0.517	0.9944
2.500	-0.718	30.000	-0.967	4.500	0.3762	2.995	0.421	0.560	0.9839

3.13. Тригонометрическая интерполяция. Дискретное преобразование Фурье и его реализация на ЭВМ

Задача интерполяции функции $y = f(x)$ заданной таблично на сетке $x_0 < x_1 < \dots < x_{n-1}$ тригонометрическим многочленом, решается во многом аналогично задаче полиномиальной интерполяции. Здесь обобщенный многочлен имеет вид

$$\Phi_n(x) = \sum_{k=0}^{n-1} a_k \exp(2\pi i k x). \quad (3.13.1)$$

В инженерных задачах широко используется представление функций в виде сходящихся тригонометрических рядов, а также различные варианты преобразования Фурье. Задача тригонометрической интерполяции имеет некоторые особенности, например, для нее оптимальным является равномерное распределение узлов.

Известно, что абсолютно интегрируемая и кусочно-гладкая на конечных отрезках функция $y = f(x)$ представима интегралом Фурье

$$f(x) = \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} f(u) \exp(2\pi i t(x-u)) du dt. \quad (3.13.2)$$

Из этого соотношения легко получить преобразования Фурье в комплексной форме:

$$\begin{cases} S(t) = \int_{-\infty}^{\infty} f(u) \exp(-2\pi i tu) du, \\ f(x) = \int_{-\infty}^{\infty} S(t) \exp(2\pi i tx) dt. \end{cases} \quad (3.13.3)$$

Эти преобразования играют большую роль в распознавании образов, в моделировании фильтров, описании изображений и в целом ряде других задач. Поскольку аналитическое вычисление интеграла (3.13.3) вызывает значительные трудности, большое распространение получили методы численной реализации этих интегралов. Был разработан метод дискретного преобразования Фурье, а в 1965 году Кули и Таки* предложили так называемый алгоритм «быстрого преобразования Фурье» - БПФ.

Под дискретным преобразованием Фурье понимают пару преобразований, которые устанавливают взаимосвязь между конечным числом дискретных выборок как функции $f(x)$, так и ее частотного спектра $S(t)$. Эти преобразования записываются обычно в виде

$$\begin{cases} S(t_n) = \Delta x \sum_{k=0}^{2N-1} f(x_k) \exp(-2\pi i t_n x_k), \quad n = 0, 1, \dots, 2N-1, \\ f(x_k) = \Delta t \sum_{n=0}^{2N-1} S(t_n) \exp(2\pi i t_n x_k), \quad k = 0, 1, \dots, 2N-1. \end{cases} \quad (3.13.4)$$

Операцию преобразования набора значений $f(x_0), f(x_1), \dots, f(x_{2N-1})$ в набор значений частотного спектра $S(t_0), S(t_1), \dots, S(t_{2N-1})$ принято называть прямым дискретным преобразованием Фурье, а обратную функцию – обратным дискретным преобразованием Фурье. Осуществление этих операций является важной составной частью многих алгоритмов.

Рассмотрим подробнее функцию $S(t_n)$ из уравнений (3.13.4). Оказывается, $S(t_n)$ имеет период $\frac{1}{\Delta x}$, что легко проверяется непосредственно:

$$S\left(t + \frac{1}{\Delta x}\right) = \Delta x \sum_{k=0}^{2N-1} f(x_k) \exp\left(-2\pi i \left(t + \frac{1}{\Delta x}\right) k \Delta x\right) = \Delta x \sum_{k=0}^{2N-1} f(x_k) \exp(-2\pi i t k \Delta x) = S(t).$$

Здесь положено $x_0 = 0$ и $x_k = k \Delta x$. Из периодичности следует, что функция $S(t)$ определяется заданием ее значений на промежутке длиной $\frac{1}{\Delta x}$, то есть выборки t_n в соотношениях (3.13.4) должны удовлетворять условию $|t_n| < 1/2\Delta x$ для всех $n = 0, 1, \dots, 2N-1$.

Если $t_n \in [0, \frac{1}{\Delta x}] = T$ и $S(t_n) = 0$, когда $t_n \notin T$, то интервал между узлами Δx необходимо выбирать из условия

$$2T\Delta x < 1. \quad (3.13.5)$$

Тогда функция $f(x)$ имеет ограниченный спектр частот, и возможно взаимное восстановление функций $f(x)$ и $S(t)$ по их значениям в конечном числе точек.

3.14. Матричная форма записи дискретного преобразования Фурье (ДПФ)

Будем предполагать далее, что исходная функция $f(x)$ имеет ограниченный спектр частот $|t_n| < T$; кроме того предположим, что задан конечный интервал изменения переменной x : $0 \leq x \leq X$.

Очевидно, $\Delta x = \frac{X}{2N}$. Из условия (3.13.5) $2T\Delta x < 1$ получаем условие на выбор числа

N : $2T\Delta x = 2T \cdot \frac{X}{2N} < 1$, $N > T \cdot X$. Отсюда же $T < \frac{N}{X}$, $\frac{2N-1}{X} \leq 2T < \frac{2N}{X}$. Положив $\Delta t = \frac{1}{X}$,

* Кули (Cooley J.W.), Таки (Tukey J.W.) – американские математики.

получим $(2N-1)\Delta t \leq 2T < \frac{2N}{X}$ и $\Delta t < \frac{1}{X} \cdot \frac{2N}{2N-1}$, где $2N-1$ - число интервалов по множеству T . Следовательно, $x_k = k\Delta x = k \cdot \frac{X}{2N}$, $t_n = n\Delta t = \frac{n}{X}$.

Тогда дискретное преобразование Фурье в точках x_k и t_n запишется в виде

$$\begin{cases} S(t_n) = \frac{X}{2N} \sum_{k=0}^{2N-1} f(x_k) \exp\left(-\frac{\pi}{N} i nk\right), & n = 0, 1, \dots, 2N-1, \\ f(x_k) = \frac{1}{X} \sum_{n=0}^{2N-1} S(t_n) \exp\left(\frac{\pi}{N} i nk\right), & k = 0, 1, \dots, 2N-1. \end{cases} \quad (3.14.1)$$

Каждое из равенств (3.14.1) можно легко получить из другого. Например обозначим $u_k = f(x_k)$ и $v_n = S(t_n)$ и умножим n -е уравнение для $S(t_n)$ на $e^{\frac{i\pi}{N} j n}$. Получим $v_n e^{\frac{i\pi}{N} j n} = \frac{X}{2N} \sum_{k=0}^{2N-1} u_k \left(e^{-\frac{i\pi}{N} n}\right)^k \cdot \left(e^{\frac{i\pi}{N} j}\right)^n$, $n = 0, 1, \dots, 2N-1$. При сложении всех $2N$ равенств легко вычисляются коэффициенты при u_k ; действительно,

$$\frac{X}{2N} \sum_{n=0}^{2N-1} e^{-\frac{i\pi}{N} nk} e^{\frac{i\pi}{N} j n} = \frac{X}{2N} \sum_{n=0}^{2N-1} e^{-i\frac{\pi}{N}(k-j)} = \begin{cases} 0, & k \neq j, \\ X, & k = j. \end{cases}$$

Пусть, например, $k \neq j$ и $k-j=1$, тогда $\sum_{n=0}^{2N-1} e^{-i\frac{\pi}{N}} = 1 + e^{-i\frac{\pi}{N}} + e^{-2i\frac{\pi}{N}} + \dots + e^{-i\frac{\pi(2N-1)}{N}} =$

$$= 1 \cdot \frac{1 - e^{-i\frac{\pi}{N} 2N}}{1 - e^{-i\frac{\pi}{N}}} = \frac{1 - e^{-2\pi i}}{1 - e^{-i\frac{\pi}{N}}} = 0. \quad \text{Разделив найденный коэффициент на } X, \text{ получим}$$

$$u_k = \frac{1}{X} \sum_{n=0}^{2N-1} v_n e^{\frac{i\pi}{N} nk}, \text{ то есть второе соотношение системы (3.14.1).}$$

Дискретные преобразования Фурье можно записать еще короче в матричном виде.

$$\text{Пусть } q = e^{-\frac{i\pi}{N}} \text{ и } U = \begin{pmatrix} u_0 \\ u_1 \\ \vdots \\ u_{2N-1} \end{pmatrix} = \begin{pmatrix} f(x_0) \\ f(x_1) \\ \vdots \\ f(x_{2N-1}) \end{pmatrix}, V = \begin{pmatrix} v_0 \\ v_1 \\ \vdots \\ v_{2N-1} \end{pmatrix} = \begin{pmatrix} S(t_0) \\ S(t_1) \\ \vdots \\ S(t_{2N-1}) \end{pmatrix},$$

$$W = \begin{pmatrix} 1 & 1 & 1 & \cdots & 1 \\ 1 & q & q^2 & \cdots & q^{2N-1} \\ 1 & q^2 & q^4 & \cdots & q^{2(2N-1)} \\ \cdots & \cdots & \cdots & \cdots & \cdots \\ 1 & q^{2N-1} & q^{2(2N-1)} & \cdots & q^{(2N-1)^2} \end{pmatrix}.$$

Тогда систему (3.14.1) можно переписать в следующем виде:

$$\begin{cases} V = \frac{X}{2N} WU, \\ U = \frac{1}{X} W^{-1}V. \end{cases} \quad (3.14.2)$$

Здесь $W^{-1} = \begin{pmatrix} 1 & 1 & 1 & \cdots & 1 \\ 1 & q^{-1} & q^{-2} & \cdots & q^{-2N-1} \\ 1 & q^{-2} & q^{-4} & \cdots & q^{-2(2N-1)} \\ \cdots & \cdots & \cdots & \cdots & \cdots \\ 1 & q^{-2N-1} & q^{-2(2N-1)} & \cdots & q^{-(2N-1)^2} \end{pmatrix}$. Очевидно из (3.14.2), что для непосредственного перемножения матриц W и U требуется $2N(2N-1) \approx (2N)^2$ операций комплексного умножения и сложения.

Разработанный Кули и Таки алгоритм БПФ использует факт, что, если число $2N$ не является простым, количество арифметических операций, требуемых для вычисления по формулам (3.14.2), можно существенно уменьшить. Если $2N = r_1 \cdot r_2 \cdots r_n$, где r_k - целые числа, то матрицу W можно представить в виде произведения n матриц W_1, W_2, \dots, W_n порядка $2N \times 2N$ таких, что у каждой матрицы W_k имеется только $r_k \cdot 2N$ отличных от нуля элементов. Тогда

$$V = \frac{X}{2N} \cdot W_n \cdot W_{n-1} \cdot \dots \cdot W_2 \cdot W_1 \cdot U \quad (3.14.3)$$

Если теперь найти алгоритм, по которому умножение матриц W_k осуществляется только путем $r_k \cdot 2N$ операций (отбрасываются умножения на нули), то для выполнения преобразования (3.14.3) потребуется $(r_1 + r_2 + \dots + r_n) \cdot 2N$ операций. Это число может быть при больших значениях $2N$ значительно меньше, чем исходное число операций $(2N)^2$. Особенностью эффективным этот алгоритм является тогда, когда $2N$ является степенью числа два. В этом случае вместо $(2N)^2$ операций требуется выполнить лишь $2N \log_2 2N$ операций. Например, для $2N = 1024 = 2^{10}$ этот алгоритм позволяет ускорить вычисления в $\frac{2N}{2N \log_2 2N} = \frac{1024}{20} \approx 50$ раз. Программы, реализующие различные варианты этого алгоритма, входят в стандартное математическое обеспечение ЭВМ и доступны массовому пользователю.

3.15. Алгоритм реализации ДПФ

Рассмотрим один из способов разложения матрицы W в случае $2N = 2^n$ и приведем алгоритм для выполнения соответствующих вычислений. Разложим матрицу W на n сомножителей W_1, W_2, \dots, W_n таких, что каждая из матриц $W_k, k = 1, 2, \dots, n$ содержит 2^{n+1} отличных от нуля элементов. Пусть $c_{i,j}^{(m)}$ - элементы матрицы W_m , то есть $W_m = \{c_{i,j}^{(m)}\}, i, j = 1, 2, \dots, 2^n$. Тогда один из алгоритмов дискретного преобразования Фурье, не изменяющий порядок чередования компонент результирующего вектора $V = \frac{X}{2N} \cdot W \cdot U$, определяется следующим образом:

$$\left\{ \begin{array}{l} q = e^{-i\frac{\pi}{N}}, i = 0, 1, \dots, 2^{n-m} - 1, \mu = 1, 2, \dots, 2^{m-1}, \\ c_{i \cdot 2^m + \mu, i \cdot 2^m - 1 + \mu}^{(m)} = 1, c_{i \cdot 2^m + 2^m - 1 + \mu, i \cdot 2^m - 1 + \mu}^{(m)} = 1, \\ c_{i \cdot 2^m + \mu, 2^{n-1} + i \cdot 2^{m-1} + \mu}^{(m)} = q^{(\mu-1)2^{n-m}}, \\ c_{i \cdot 2^m + 2^m - 1 + \mu, 2^{n-1} + i \cdot 2^{m-1} + \mu}^{(m)} = -q^{(\mu-1)2^{n-m}}, \end{array} \right. \quad (3.15.1)$$

$c_{i,j}^{(m)} = 0$ для всех остальных пар индексов i, j .

Для определяемых таким образом матриц W_m легко указать алгоритм последовательного выполнения операций при умножении $W_m \cdot W_{m-1} \cdots W_1 \cdot U$. Пусть $\delta_r^{(1)}$, $r = 0, 1, \dots, 2^n - 1$ - компоненты вектора-столбца $W_1 \cdot U$. Тогда для каждого $i = 0, 1, \dots, 2^{n-1} - 1$ они вычисляются следующим образом:

$$\begin{cases} \delta_{2i}^{(1)} = U_i + U_{2^{n-1}+i}, \\ \delta_{2i+1}^{(1)} = U_i - U_{2^{n-1}+i}. \end{cases} \quad (3.15.2)$$

Далее, после умножения слева на W_2 компоненты $\delta_r^{(2)}$, $r = 0, 1, \dots, 2^n - 1$ вектора-столбца $W_2 \cdot W_1 \cdot U$ при каждом $i = 0, 1, \dots, 2^{n-2} - 1$ и для $\mu = 0, 1$ вычисляются с помощью равенств

$$\begin{cases} \delta_{i \cdot 2^2 + \mu}^{(2)} = \delta_{2i+\mu}^{(1)} + q^{i \cdot 2^{n-2}} \delta_{2^{n-1}+2i+\mu}^{(1)}, \\ \delta_{i \cdot 2^2 + 2 + \mu}^{(2)} = \delta_{2i+\mu}^{(1)} - q^{i \cdot 2^{n-2}} \delta_{2^{n-1}+2i+\mu}^{(1)}. \end{cases} \quad (3.15.3)$$

Формулы (3.15.2) и (3.15.3) для произвольного $m = 3, 4, \dots, n$ обобщаются аналогично. Действительно, пусть $\delta_r^{(m)}$, $r = 0, 1, \dots, 2^n - 1$ - компоненты вектора-столбца $W_m \cdot W_{m-1} \cdots W_1 \cdot U$. Тогда для каждого $i = 0, 1, \dots, 2^{n-m} - 1$ и для всех $\mu = 0, 1, \dots, 2^{m-1}$ имеем равенства

$$\begin{cases} \delta_{i \cdot 2^m + \mu}^{(m)} = \delta_{2^{m-1} \cdot i + \mu}^{(m-1)} + q^{i \cdot 2^{n-m}} \delta_{2^{n-1} + 2^{m-1} \cdot i + \mu}^{(m-1)}, \\ \delta_{i \cdot 2^m + 2^{m-1} + \mu}^{(m)} = \delta_{2^{m-1} \cdot i + \mu}^{(m-1)} - q^{i \cdot 2^{n-m}} \delta_{2^{n-1} + 2^{m-1} \cdot i + \mu}^{(m-1)}. \end{cases} \quad (3.15.4)$$

Последовательный переход от вектора-столбца $W_{m-1} \cdots W_1 \cdot U$ к вектору-столбцу $W_m \cdot W_{m-1} \cdots W_1 \cdot U$ требует выполнения 2^n основных операций, поэтому для вычисления компонент вектора V требуется произвести $n \cdot 2^n$ основных операций, то есть применение алгоритма БПФ сокращает число операций в $2^n / n$ раз.

3.16. Пример реализации алгоритма ДПФ при $2N = 4$

Рассмотрим для наглядности самый ростой случай: $2^n = 4$, $n = 2$. Здесь $\frac{2N}{2 \log_2 2N} = \frac{4}{2 \log_2 4} = 1$, то есть применение БПФ не дает никаких преимуществ по сравнению с формулами (3.14.2). Однако этот простейший пример позволяет проиллюстрировать и понять сущность алгоритма ДПФ, то есть формул (3.15.1) - (3.15.4).

$2N = 2^n = 4$, $n = 2$, $q = e^{-i\pi/2}$, $q^2 = -1$, $q^3 = -q$, $q^4 = 1$, $q^5 = q$, $q^6 = -1$, $q^7 = -q$, $q^8 = 1$, $q^9 = q$. Матрица W разлагается всего на две матрицы: $W = W_2 \cdot W_1$ и имеет вид

$$W = \begin{pmatrix} 1 & 1 & 1 & 1 \\ 1 & q & q^2 & q^3 \\ 1 & q^2 & q^4 & q^6 \\ 1 & q^3 & q^6 & q^9 \end{pmatrix} = \begin{pmatrix} 1 & 1 & 1 & 1 \\ 1 & q & -1 & -q \\ 1 & -1 & 1 & -1 \\ 1 & -q & -1 & q \end{pmatrix}.$$

Обе матрицы W_1 и W_2 содержат $2^{n+1} = 2^3 = 8$ ненулевых элементов. Найдем их по формулам (3.15.1).

$$m = 1, i = 0, 1, \dots, 2^{n-m} - 1 = 0, 1,$$

$$\mu = 1, 2, \dots, 2^{m-1} = 1.$$

$$\begin{aligned}
& i=0, \mu=1. c_{1,1}^{(1)}=1, c_{1+1,1}^{(1)}=c_{2,1}^{(1)}=1, \\
& c_{1,2+1}^{(1)}=c_{1,3}^{(1)}=1, c_{1+1,2+1}^{(1)}=c_{2,3}^{(1)}=-1, \text{ так как } q^{(\mu-1)2^{n-m}}=q^0=1, \\
& i=0, \mu=1. c_{2+1,1+1}^{(1)}=c_{3,2}^{(1)}=1, c_{2+1+1,1+1}^{(1)}=c_{4,2}^{(1)}=1, \\
& c_{2+1,2+1+1}^{(1)}=c_{3,4}^{(1)}=1, c_{2+1+1,2+1+1}^{(1)}=c_{4,4}^{(1)}=-1.
\end{aligned}$$

Таким образом, $W_1 = \begin{pmatrix} 1 & 0 & 1 & 0 \\ 1 & 0 & -1 & 0 \\ 0 & 1 & 0 & 1 \\ 0 & 1 & 0 & -1 \end{pmatrix}$, так как все остальные $c_{i,j}^{(1)} = 0$.

$$m=2, i=0,1,\dots,2^{n-m}-1=0,$$

$$\mu=1,2,\dots,2^{m-1}=1,2.$$

$$\begin{aligned}
& i=0, \mu=1. c_{1,1}^{(2)}=1, c_{2+1,1}^{(2)}=c_{3,1}^{(2)}=1, \\
& c_{1,2+1}^{(2)}=c_{1,3}^{(2)}=1, c_{2+1,2+1}^{(2)}=c_{3,3}^{(2)}=-1,
\end{aligned}$$

$$\begin{aligned}
& i=0, \mu=2. q^{(\mu-1)2^{n-m}}=q, c_{2,2}^{(2)}=1, c_{2+2,2}^{(2)}=c_{4,2}^{(2)}=1, \\
& c_{2,2+2}^{(2)}=c_{2,4}^{(2)}=q, c_{2+2,2+2}^{(2)}=c_{4,4}^{(2)}=-q.
\end{aligned}$$

Итак, $W_2 = \begin{pmatrix} 1 & 0 & 1 & 0 \\ 0 & 1 & 0 & q \\ 1 & 0 & -1 & 0 \\ 0 & 1 & 0 & -q \end{pmatrix}$.

$$\text{Тогда } W = W_2 \cdot W_1 = \begin{pmatrix} 1 & 0 & 1 & 0 \\ 0 & 1 & 0 & q \\ 1 & 0 & -1 & 0 \\ 0 & 1 & 0 & -q \end{pmatrix} \cdot \begin{pmatrix} 1 & 0 & 1 & 0 \\ 1 & 0 & -1 & 0 \\ 0 & 1 & 0 & 1 \\ 0 & 1 & 0 & -1 \end{pmatrix} = \begin{pmatrix} 1 & 1 & 1 & 1 \\ 1 & q & -1 & -q \\ 1 & -1 & 1 & -1 \\ 1 & -q & -1 & q \end{pmatrix}.$$

Найдем $\delta_r^{(1)}$, $r=0,1,\dots,2^n-1=0,1,2,3$ ненулевые компоненты вектора-столбца $W_1 \cdot U$:

$$W_1 \cdot U = \begin{pmatrix} 1 & 0 & 1 & 0 \\ 1 & 0 & -1 & 0 \\ 0 & 1 & 0 & 1 \\ 0 & 1 & 0 & -1 \end{pmatrix} \cdot \begin{pmatrix} u_0 \\ u_1 \\ u_2 \\ u_3 \end{pmatrix} = \begin{pmatrix} u_0 + u_2 \\ u_0 - u_2 \\ u_1 + u_3 \\ u_1 - u_3 \end{pmatrix}. \quad \text{Легко видеть, что если положить}$$

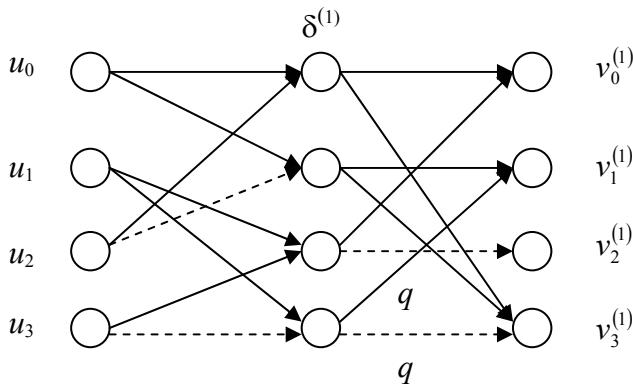
$i=0,1,\dots,2^{n-1}-1=0,1$, то компоненты $W_1 \cdot U$ вычисляются по формулам (3.15.2). Аналогично,

$$W_2 \cdot W_1 \cdot U = \begin{pmatrix} 1 & 0 & 1 & 0 \\ 0 & 1 & 0 & q \\ 1 & 0 & -1 & 0 \\ 0 & 1 & 0 & -q \end{pmatrix} \cdot \begin{pmatrix} \delta_0^{(1)} \\ \delta_1^{(1)} \\ \delta_2^{(1)} \\ \delta_3^{(1)} \end{pmatrix} = \begin{pmatrix} \delta_0^{(1)} + \delta_2^{(1)} \\ \delta_1^{(1)} + q\delta_3^{(1)} \\ \delta_0^{(1)} - \delta_2^{(1)} \\ \delta_1^{(1)} - q\delta_3^{(1)} \end{pmatrix}, \text{ то есть вычисление ведется строго по}$$

формулам (3.15.3).

В результате этих вычислений получим вектор $V^{(1)}$, отличающийся от искомого множителем $\frac{4}{X}$. Его компоненты записываются следующим образом:

$$\begin{aligned}
 v_0^{(1)} &= \delta_0^{(1)} + \delta_2^{(1)} = (u_0 + u_2) + (u_1 + u_3), \\
 v_1^{(1)} &= \delta_1^{(1)} + q\delta_3^{(1)} = (u_0 - u_2) + q(u_1 - u_3), \\
 v_2^{(1)} &= \delta_0^{(1)} - \delta_2^{(1)} = (u_0 + u_2) - (u_1 + u_3), \\
 v_3^{(1)} &= \delta_1^{(1)} - q\delta_3^{(1)} = (u_0 - u_2) - q(u_1 - u_3).
 \end{aligned}$$



как только $N > 2$, будет достигнута экономия в числе операций, например, при $2N = 8$

$$\frac{2N}{2\log_2 2N} = \frac{8}{2\log_2 8} = \frac{8}{6} = 1.33$$
, то есть экономия составит примерно 30%.

Граф соответствующих вычислений компонент векторов $\delta^{(1)} = W_1 \cdot U$ и $V^{(1)} = W_2 \cdot \delta^{(1)}$ изображен слева. Сплошные стрелки на графике означают сложение, пунктирные – вычитание. Значок q на стрелке означает, что слагаемое (вычитаемое), соответствующее концу стрелки, умножается предварительно на множитель q . Аналогично можно изобразить вариант реализации ДПФ для $2N = 8$ и так далее. При этом

3.17. Лабораторная работа № 7. Дискретное преобразование Фурье

Mathcad содержит функции для выполнения быстрого дискретного преобразования Фурье (БПФ) и его обращения. Таких функций восемь (Fourier Transform): **CFFT(A)**, **cfft(A)**, **FFT(v)**, **fft(v)**, **ICFFT(A)**, **icfft(A)**, **IFFT(v)**, **ifft(v)**. Все функции берут в качестве аргументов и возвращают векторы и матрицы. Чаще всего используются две группы подпрограмм.

I. Функции **fft** и **ifft**, если выполнены следующие два условия:

а) аргументы вещественны и б) вектор данных имеет 2^n элементов. В этом случае вторая половина преобразования Фурье является комплексно сопряженной с первой. Mathcad отбрасывает вторую половину вектора–результата. Это сохраняет время и память при вычислениях.

Второе условие позволяет использовать высоко эффективный алгоритм быстрого преобразования Фурье. Различные формулировки определения преобразования Фурье используют различные нормировочные коэффициенты и соглашения о знаке перед мнимой единицей в показателе экспоненты прямого и обратного преобразований. Функции **fft**, **ifft**, **cfft** и **icfft** используют $\frac{1}{\sqrt{2N}}$ в качестве нормировочного коэффициента и положительный

показатель степени в прямом преобразовании. Функции **FFT**, **IFFT**, **CFFT** и **ICFFT** употребляют величину $\frac{1}{2N}$ в качестве нормировочного коэффициента и отрицательный показатель степени в прямом преобразовании. Функции **fft**, **ifft**, **cfft** и **icfft** применяются только попарно.

II. Функции **cfft** и **icfft** не используют симметрию в преобразовании и поэтому применяются для комплексных данных. Вектор данных в этом случае может иметь размерность, отличную от 2^n . Пара преобразований **cfft**, **icfft** работает значительно быстрее, когда число строк и столбцов массива данных может быть представлено в виде произведения большого числа меньших сомножителей (см. формулу (3.14.3)). С другой стороны, вектор, длина которого – большое простое число, заметно замедлит вычисление преобразования Фурье. Когда в качестве аргумента **cfft** используется матрица, результат есть двумерное преобразование Фурье исходной таблицы.

Рассмотрим подробнее преобразование Фурье в вещественной области. Функция **fft(v)** возвращает дискретное преобразование Фурье 2^n -мерного вещественнозначного вектора. Аргумент можно интерпретировать как результат измерений через равные промежутки времени некоторого сигнала. Результат работы **fft(v)** – комплексный вектор длиной $2^{n-1} + 1$. Если вектор v имеет размерность, отличную от 2^n , Mathcad выдает сообщение об ошибке: «неверный размер вектора».

Элементы вектора, возвращаемого **fft**, вычисляются по формуле

$$v_n = \frac{1}{\sqrt{2N}} \sum_{k=0}^{2N-1} u_k \exp\left(i \frac{n}{N} \pi k\right), \quad n = 0, 1, \dots, N+1, \quad 2N = 2^n.$$

Это первая формула в системе (3.14.1), отличающаяся от первоначальной только знаком мнимой единицы в показателе экспоненты. Элементы вектора, возвращенные функцией **fft**, соответствуют различным частотам. Чтобы восстановить фактическую частоту, необходимо знать частоту измерения исходного сигнала. Если частота измерения исходного сигнала ω , то частота, соответствующая v_n равна:

$$\omega_n = \frac{n}{2N} \omega.$$

Функция **ifft(v)** возвращает обратное дискретное преобразование Фурье; результат ее работы – вещественнозначный. Вектор v должен иметь $2^{n-1} + 1$ элемент, размерность результата 2^n . Если размерности не совпадают со стандартом, выдается аналогичное сообщение об ошибке. Чтобы вычислить результат, Mathcad сначала создает новый вектор w , комплексно сопряженный с v , и присоединяет его к вектору v . Затем вычисляет вектор u по формуле

$$u_k = \frac{1}{\sqrt{2N}} \sum_{n=0}^{2N-1} \tilde{v}_n \exp\left(i \frac{\pi}{N} nk\right), \quad k = 0, 1, \dots, 2N-1, \quad \tilde{v} = v + w.$$

Функции **fft** и **ifft** – точные обращения. Для всех вещественнозначных v справедливо $\text{ifft}(\text{fft}(v))=v$.

Данные выше определения преобразования Фурье не являются единственными возможными. Результат действия функций **FFT**, **IFFT**, **CFFT** и **ICFFT** отличается следующим.

1. Вместо коэффициента $\frac{1}{\sqrt{2N}}$ перед обеими формулами в прямом преобразовании

стоит коэффициент $\frac{1}{2N}$ и коэффициент единица в обратном преобразовании.

2. Знак минус появляется в показателе экспоненты прямого преобразования и исчезает в формуле обратного.

Таким образом, в этих функциях реализуется следующая система формул:

$$\begin{cases} v_n = \frac{1}{2N} \sum_{n=0}^{2N-1} u_k \exp\left(i \frac{\pi}{N} nk\right), & n = 0, 1, \dots, N-1, \\ u_k = \sum_{n=0}^{2N-1} \tilde{v}_n \exp\left(i \frac{\pi}{N} nk\right), & k = 0, 1, \dots, 2N-1. \end{cases} \quad (3.17.1)$$

Приведенные формулы практически полностью аналогичны системе (3.14.1).

При практических вычислениях по формулам (3.17.1) следует помнить, что первый элемент векторов u и w должен иметь нулевой индекс, а не первый из-за разных значений комплексной экспоненты. В программах **fft**, **ifft**, **FFT** и **IFFT** это учтено, если элементы v_0 и w_0 не определены, Mathcad автоматически устанавливает их равными нулю. Неучет этого обстоятельства может привести к искажению результата.

Вычислим дискретное преобразование Фурье непосредственно по формулам (3.17.1) и с помощью функций **FFT** и **IFFT**:

ORIGIN := 0 TOL := 10⁻⁹

$$X1 := \begin{pmatrix} 1.000 \\ 1.803 \\ 3.085 \\ 4.778 \\ 6.434 \\ 7.347 \\ 7.027 \\ 5.652 \end{pmatrix} \quad X2 := \begin{pmatrix} 3.897 \\ 2.381 \\ 1.347 \\ 7.422 \\ 0.419 \\ 0.256 \\ 0.176 \\ 0.142 \end{pmatrix} \quad pi := 3.14159 \quad N := 8 \quad ind := 2 * N - 1 \quad u := \text{stack}(X1, X2)$$

$$n := 0 \dots \text{ind} \quad v_n = \frac{1}{\text{ind} + 1} \cdot \left[\sum_{k=0}^{\text{ind}} \left(u_k \cdot \exp\left(-i \cdot \frac{pi}{N} \cdot n \cdot k\right) \right) \right]$$

$$k := 0 \dots \text{ind} \quad x_k = \sum_{n=0}^{\text{ind}} \left(v_n \cdot \exp\left(-i \cdot \frac{pi}{N} \cdot n \cdot k\right) \right)$$

$$u = \begin{pmatrix} 1 \\ 1.803 \\ 3.085 \\ 4.778 \\ 6.434 \\ 7.347 \\ 7.027 \\ 5.652 \\ 3.897 \\ 2.381 \\ 1.347 \\ 7.422 \\ 0.419 \\ 0.256 \\ 0.176 \\ 0.142 \end{pmatrix} \quad v = \begin{pmatrix} 3.323 \\ -0.991 - 1.13i \\ -0.556 + 0.041i \\ 0.461 - 0.182i \\ 7.181 \cdot 10^{-3} + 0.388i \\ -0.372 - 0.175i \\ 0.312 - 0.305i \\ 0.177 + 0.381i \\ -0.4 - 1.24 \cdot 10^{-5}i \\ 0.177 - 0.381i \\ 0.312 + 0.305i \\ -0.372 + 0.175i \\ 7.206 \cdot 10^{-3} - 0.388i \\ 0.461 + 0.182i \\ -0.556 - 0.041i \\ -0.991 + 1.13i \end{pmatrix} \quad x = \begin{pmatrix} 1 - 5.426 \cdot 10^{-5}i \\ 1.803 - 4.544 \cdot 10^{-5}i \\ 3.085 - 3.663 \cdot 10^{-5}i \\ 4.778 - 2.781 \cdot 10^{-5}i \\ 6.434 - 1.899 \cdot 10^{-5}i \\ 7.347 - 1.017 \cdot 10^{-5}i \\ 7.027 - 1.356 \cdot 10^{-6}i \\ 5.652 + 7.462 \cdot 10^{-6}i \\ 3.897 + 1.628 \cdot 10^{-5}i \\ 2.381 + 2.51 \cdot 10^{-5}i \\ 1.347 + 3.391 \cdot 10^{-5}i \\ 7.422 + 4.273 \cdot 10^{-5}i \\ 0.419 + 5.155 \cdot 10^{-5}i \\ 0.256 + 6.037 \cdot 10^{-5}i \\ 0.176 + 6.918 \cdot 10^{-5}i \\ 0.142 + 7.8 \cdot 10^{-5}i \end{pmatrix}$$

При наборе в пакете Mathcad формул для v_n и x_k следует помнить, что для ввода мнимого числа нужно вслед за его модулем ввести символ мнимой единицы i или j , например, $1i$ или $2.5j$. Нельзя использовать i или j сами по себе для обозначения мнимой единицы. Нужно всегда набирать $1i$ или $1j$, в противном случае Mathcad истолкует i или j как переменные. Когда курсор покидает выражение, содержащее $1i$, Mathcad скрывает избыточную единицу.

Вычислим ДПФ теперь с помощью встроенных функций:

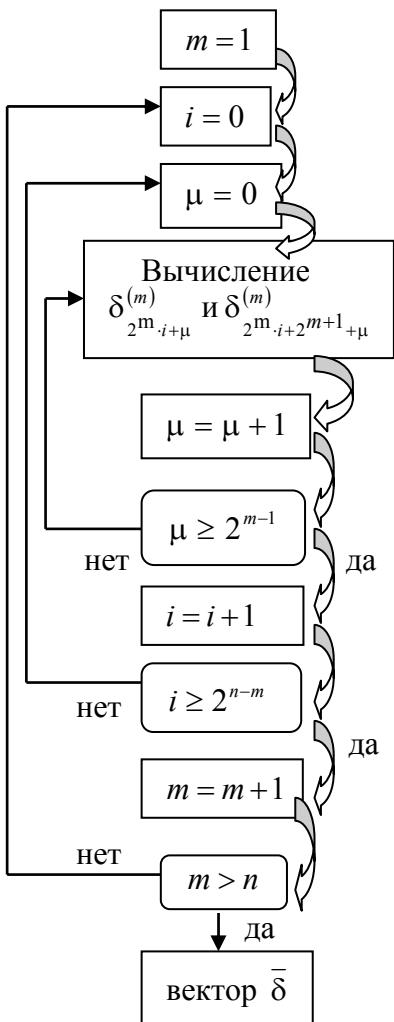
$$v1 := FFT(u) \quad x1 := IFFT(v1)$$

Значения векторов u , $v1$ и $x1$ приведены на следующей странице. Видно, что вторая половина вектора v является комплексно сопряженной с первой. Программа **FFT** учитывает это обстоятельство; у вектора $v1$ вторая половина уже отброшена. Ненулевая мнимая часть компонент вектора x объясняется лишь недостаточной точностью вычислений.

Программа, реализующая вычисление очередного m -го вектор-столбца $W_m \cdot W_{m-1} \cdot \dots \cdot W_1 \cdot U$ по формулам (3.15.4) и ее блок-схема приводятся ниже. Комплексная экспонента в программе рассчитывается по формуле Эйлера $e^{ix} = \sin x + i \cos x$. В приведенной подпрограмме FourierTransform параметр v определяет исходный вектор данных длиной 2^n элементов, параметр $kind$ задает либо прямое преобразование ($kind=-1$), либо обратное ($kind=1$).

$$u = \begin{pmatrix} 1 \\ 1.803 \\ 3.085 \\ 4.778 \\ 6.434 \\ 7.347 \\ 7.027 \\ 5.652 \\ 3.897 \\ 2.381 \\ 1.347 \\ 7.422 \\ 0.419 \\ 0.256 \\ 0.176 \\ 0.142 \end{pmatrix} \quad v1 = \begin{pmatrix} 1 \\ 1.803 \\ 3.085 \\ 4.778 \\ 6.434 \\ 7.347 \\ 7.027 \\ 5.652 \\ 3.897 \\ 2.381 \\ 1.347 \\ 7.422 \\ 0.419 \\ 0.256 \\ 0.176 \\ 0.142 \end{pmatrix} \quad x1 = \begin{pmatrix} 1 \\ 1.803 \\ 3.085 \\ 4.778 \\ 6.434 \\ 7.347 \\ 7.027 \\ 5.652 \\ 3.897 \\ 2.381 \\ 1.347 \\ 7.422 \\ 0.419 \\ 0.256 \\ 0.176 \\ 0.142 \end{pmatrix}$$

$$v = \begin{pmatrix} 3.323 \\ -0.991 - 1.13i \\ -0.556 + 0.041i \\ 0.461 - 0.182i \\ 7.188 \cdot 10^{-3} + 0.388i \\ -0.372 - 0.175i \\ 0.312 - 0.305i \\ 0.177 + 0.381i \\ -0.4 \end{pmatrix}$$



FourierTransform(v, kind) :=

$n4 \leftarrow \text{length}(v)$ $n \leftarrow \frac{\ln(n4)}{\ln(2)}$ $n1 \leftarrow n4 - 1$ $\text{for } m \in 0..n1$ $A_m \leftarrow \text{Re}(v_m)$ $B_m \leftarrow \text{Im}(v_m)$ $\text{for } m \in 1..n$ $n2 \leftarrow 2^m$ $n3 \leftarrow \frac{n2}{2}$ $\text{for } i \in 0..2^{n-m} - 1$ $\text{for } \mu \in 0..n3 - 1$ $j \leftarrow n2 \cdot i + \mu$ $l \leftarrow n3 \cdot i + \mu$ $c \leftarrow 3.141593 \cdot \frac{\mu}{n3}$ $si \leftarrow \text{if}(kind < 0, \sin(c), -\sin(c))$ $co \leftarrow \cos(c)$ $ni \leftarrow 2^{n-1} + 1$ $jm \leftarrow n3 + j$ $ass \leftarrow A_{ni}$

```

    bob←Bni
    A2←ass·si
    A1←ass·co
    B1←bob·si
    B2←bob·co
    AAj←A1+A1+B1
    BBj←B1-A2+B2
    AAjm←A1-A1-B1
    BBjm←B1+A2-B2
    for 1∈0..n1
        A1←AA1
        B1←BB1
    for 1∈0..n1      if kind<0
        AA1←AA1/n4
        BB1←BB1/n4
    i←√-1
    v←AA+BB·i
    v

```

Вычислим прямое и обратное преобразование Фурье для начального вектора u с помощью данной подпрограммы.

$v5 := \text{FourierTransform}(u, -1)$ $v6 := \text{FourierTransform}(v5, 1)$

$$u = \begin{pmatrix} 1 \\ 1.803 \\ 3.085 \\ 4.778 \\ 6.434 \\ 7.347 \\ 7.027 \\ 5.652 \\ 3.897 \\ 2.381 \\ 1.347 \\ 7.422 \\ 0.419 \\ 0.256 \\ 0.176 \\ 0.142 \end{pmatrix}, \quad v5 = \begin{pmatrix} 3.323 \\ -0.991-1.13i \\ -0.556+0.041i \\ 0.461-0.182i \\ 7.181 \cdot 10^{-3} + 0.388i \\ -0.372-0.175i \\ 0.312-0.305i \\ 0.177+0.381i \\ -0.4 \\ 0.177-0.381i \\ 0.312+0.305i \\ -0.372+0.175i \\ 7.206 \cdot 10^{-3}-0.388i \\ 0.461+0.182i \\ -0.556-0.041i \\ -0.991+1.13i \end{pmatrix}, \quad v6 = \begin{pmatrix} 1 \\ 1.803 \\ 3.085 \\ 4.778 \\ 6.434 \\ 7.347 \\ 7.027 \\ 5.652 \\ 3.897 \\ 2.381 \\ 1.347 \\ 7.422 \\ 0.419 \\ 0.256 \\ 0.176 \\ 0.142 \end{pmatrix}$$

Задание № 1. При помощи любой программы, разобранной в этой лабораторной работе, вычислить прямое и обратное преобразование Фурье для заданного начального вектора.

№ варианта	Исходный вектор u .
1	7.38; 6.76; 5.22; 3.47; 2.08; 1.16; 0.64; 0.36; 0.23; 0.16; 0.13; 0.13; 0.16; 0.25; 0.37; 0.71.
2	-1.24; -1.17; -1.08; -0.96; -0.84; -0.79; -0.80; -0.90; -1.10; -1.21; -1.02; -1.28; -1.32; -1.34; -1.36; -1.37.
3	2.312; 2.251; 2.418; 2.752; 2.459; 2.700; 3.022; 3.079; 2.420; 2.669; 3.241; 3.615; 3.591; 4.103; 4.481; 4.492.
4	-3.00; -3.58; -4.12; -5.56; -4.86; -4.99; -4.94; -4.73; -4.36; -3.86; -3.30; -2.70; -2.13; -1.64; -1.26; -1.05.
5	1.03; 1.05; 90.60; 520.40; 1714.70; 2915.00; 2439.20; 1020.60; 230.70; 32.17; 3.29; 0.30; 0.03; 0.004; 0.001; 0.0003.
6	2980.10; 2089.30; 742.40; 146.60; 18.60; 1.80; 0.16; 0.02; 0.003; 0.001; 0.001; 0.001; 0.002; 0.003; 0.018; 0.90.
7	1.08; 1.34; 1.75; 2.18; 2.53; 2.71; 2.65; 2.37; 1.97; 1.54; 1.16; 0.86; 0.64; 0.50; 0.42; 0.37.
8	2.71; 2.60; 2.28; 1.86; 1.44; 1.07; 0.80; 0.46; 0.42; 0.40; 0.37; 0.37; 0.40; 0.48; 0.60; 0.80.
9	-1.32; -1.28; -1.26; -1.24; 1.25; -1.25; -1.29; -1.21; -1.26; -1.27; -1.29; -1.28; -1.33; -1.34; -1.37; -1.39.
10	-4.09; -4.20; -4.50; -4.71; -4.93; -5.07; -4.92; -4.85; -4.60; -4.44; -4.12; -3.85; -3.50; -3.01; -3.07; -3.10.
11	1.02; 2.44; 5.42; 10.41; 16.37; 19.92; 18.64; 13.43; 7.75; 3.61; 1.69; 0.64; 0.27; 0.10; 0.53; 1.04.
12	20.05; 17.50; 11.91; 6.43; 2.97; 1.25; 0.54; 0.29; 0.10; 0.06; 0.05; 0.10; 0.51; 1.12; 1.26; 2.92.
13	-1.11; -0.81; -0.33; 0.32; 0.75; 0.84; 0.71; 0.59; 0.04; -0.67; -0.92; -1.13; -1.27; -1.32; -1.35; -1.37.
14	30.528; 34.221; 34.233; 34.114; 33.595; 34.058; 34.498; 35.822; 35.678; 37.442; 35.698; 36.742; 37.244; 38.884; 37.167; 38.696.
15	-2.07; -2.82; -3.71; -4.35; -4.76; -4.99; -4.90; -4.54; -4.10; -3.33; -2.47; -1.50; -0.62; -0.04; -0.68; 0.92.
16	1.10; 3.28; 9.51; 22.80; 41.44; 53.96; 49.42; 31.97; 15.22; 5.70; 1.80; 0.55; 0.17; 0.06; 0.03; 0.02.
17	115.18; 115.28; 115.18; 116.01; 117.20; 119.04; 121.43; 119.35; 120.84; 121.58; 124.33; 125.96; 124.72; 127.26; 125.29; 126.04.
18	-0.78; -1.22; -1.34; -1.39; -1.42; -1.43; -1.42; -1.41; -1.37; -1.30; -1.10; -0.10; 1.12; 1.25; 1.33; 1.36.
19	54.50; 45.71; 27.27; 12.13; 4.32; 1.35; 0.44; 0.13; 0.05; 0.03; 0.02; 0.13; 0.41; 1.37; 4.38; 12.10.
20	2.00; -0.06; -1.90; -3.42; -4.91; -4.87; 4.06; -2.77; -1.10; 0.95; 3.12; 5.34; 6.79; 8.10; 8.91; 7.47.
21	-0.78; 0.18; 0.89; 1.13; 1.21; 1.24; 1.23; 1.18; 1.04; 0.63; -0.38; -1.01; -1.22; -1.30; -1.35; -1.27.
22	1.10; 1.32; 1.40; 1.43; 1.45; 1.46; 1.44; 1.42; 1.37; 1.25; 0.76; -0.80; -1.22; -1.33; -1.36; -1.29.
23	1.03; 4.36; 16.71; 49.87; 105.04; 146.33; 130.97; 75.98; 30.00; 8.75; 2.11; 0.47; 0.11; 0.03; 0.01; 0.24.
24	148.41; 118.86; 62.64; 22.52; 6.21; 1.45; 0.33; 0.08; 0.02; 0.01; 0.08; 0.32; 1.45; 6.27; 22.64; 62.25.
25	0.01; 0.97; 1.23; 1.32; 1.36; 1.28; 1.13; 0.64; -0.64; -1.13; -1.28; -1.34; -1.37; -1.23; -0.91; -0.25.
26	1.03; 7.82; 51.55; 238.10; 675.94; 1075.48; 920.17; 429.33; 110.81; 20.86; 2.83; 0.35; 0.04; 0.01; -0.49; -3.68.
27	-0.01; -1.47; -2.82; -3.93; -4.65; -4.98; -4.87; -4.33; -3.47; -2.16; -0.74; 0.74; 2.17; 3.14; 4.33; 4.87.
28	1.17; 5.83; 29.38; 108.91; 266.47; 396.75; 347.18; 180.50; 59.21; 13.54; 2.44; 0.49; 0.07; 0.01; 0.03; 0.11.
29	1.05; -0.77; -2.31; -3.66; -4.64; -4.95; -4.80; -4.11; -3.08; -1.67; 0.12; 1.94; 3.66; 5.17; 6.29; 6.98.
30	0.78; 1.22; 1.34; 1.39; 1.42; 1.37; 1.30; 1.11; 0.18; -1.12; -1.27; -1.33; -1.36; -1.28; 1.17; -0.65.

4. ЧИСЛЕННОЕ ДИФФЕРЕНЦИРОВАНИЕ И ЧИСЛЕННОЕ ИНТЕГРИРОВАНИЕ

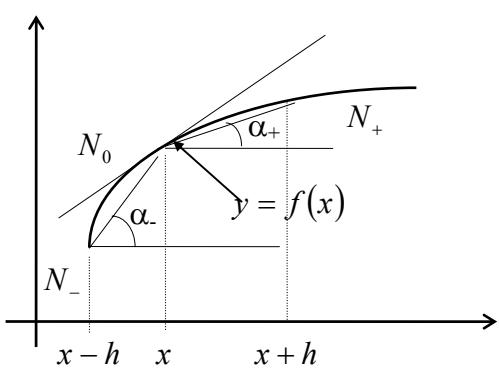
Численное дифференцирование применяется тогда, когда функцию невозможно про-дифференцировать аналитически, например, когда она задана таблично.

4.1. Простейшие формулы численного дифференцирования для первой производной

Из определения первой производной $f'(x) = \lim_{\Delta x \rightarrow 0} \frac{f(x + \Delta x) - f(x)}{\Delta x}$ естественно ис-пользовать для ее вычисления две простейшие приближенные формулы

$$f'(x) \approx \frac{f(x + h) - f(x)}{h}, \quad (4.1.1)$$

$$f'(x) \approx \frac{f(x) - f(x - h)}{h}, \quad (4.1.2)$$



соответствующие выбору фиксированных значений $\Delta x = h$ и $\Delta x = -h$. Здесь $h > 0$ - малый па-раметр - шаг. Формулы (4.1.1) и (4.1.2) называют правой и левой разностными производными. Оценим их по-грешности:

$$r_+(x, h) = f'(x) - \frac{f(x + h) - f(x)}{h} \quad \text{и}$$

$$r_-(x, h) = f'(x) - \frac{f(x) - f(x - h)}{h}, \quad \text{воспользовавшись} \\ \text{формулой Тейлора:}$$

$$f(x \pm h) = f(x) \pm f'(x)h + \frac{f''(\zeta_\pm)}{2}h^2, \quad \zeta_+ \in (x, x + h), \quad \zeta_- \in (x - h, x). \quad (4.1.3)$$

Подставив в $r_\pm(x, h)$ выражение (4.1.3), получим

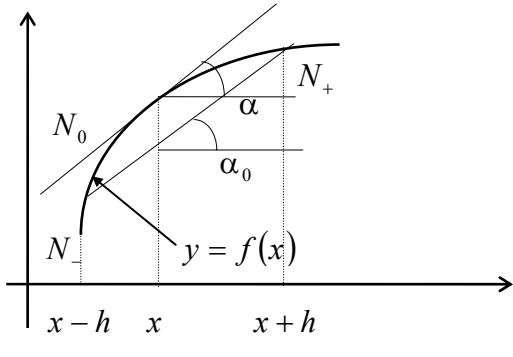
$$r_+(x, h) = f'(x) - \frac{f(x) + f'(x)h + \frac{f''(\zeta_+)}{2}h^2 - f(x)}{h} = \frac{1}{h} \left[f'(x)h - f'(x)h - \frac{f''(\zeta_+)}{2}h^2 \right] = -\frac{f''(\zeta_+)}{2}h.$$

Аналогично, $r_-(x, h) = \frac{f''(\zeta_-)}{2}h$. Таким образом,

$$\begin{cases} |r_+(x, h)| \leq \frac{1}{2}M_2h, \quad M_2 = \max_{[x, x+h]}|f''(x)|, \\ |r_-(x, h)| \leq \frac{1}{2}M_2h, \quad M_2 = \max_{[x-h, x]}|f''(x)|. \end{cases} \quad (4.1.4)$$

Итак, формулы (4.1.1) и (4.1.2) имеют первый порядок точности по h . Геометриче-ская интерпретация этих формул показана на предыдущем рисунке. Естественно предполо-жить, что лучшим по сравнению с (4.1.1) и (4.1.2) приближением $f'(x) = \tan \alpha$ является тан-генс угла наклона α_0 секущей к графику $y = f(x)$, проведенной через точки N_- и N_+ . Со-ответствующая формула приближения имеет вид

$$f'(x) \approx \frac{f(x + h) - f(x - h)}{2h}. \quad (4.1.5)$$



$f'(x)$, полученную по формуле (4.1.5), называют центральной разностной производной. Оценим опять погрешность формулы (4.1.5). Для этого подставим в выражение для погрешности $r_0(x, h) = f'(x) - \frac{f(x+h) - f(x-h)}{2h}$ соответствующие разложения в ряд Тейлора:

$$f(x \pm h) = f(x) \pm f'(x)h + \frac{f''(x)}{2}h^2 \pm \frac{f^{(3)}(\zeta_{\pm})}{6}h^3. \text{ По-}$$

лучим

$$\begin{aligned} r_0(x, h) &= f'(x) - \frac{f(x) + f'(x)h + \frac{1}{2}f''(x)h^2 + \frac{1}{6}f'''(\zeta_+)h^3 - f(x) + f'(x)h - \frac{1}{2}f''(x)h^2}{2h} + \\ &+ \frac{\frac{1}{6}f'''(\zeta_-)h^3}{2h} = f'(x) - f'(x) - \frac{h^2}{12}(f'''(\zeta_+) + f'''(\zeta_-)) = -\frac{h^2}{12}(f'''(\zeta_+) + f'''(\zeta_-)). \end{aligned}$$

Следовательно, справедлива оценка погрешности

$$|r_0(x, h)| \leq \frac{M_3}{6}h^2, M_3 = \max_{[x-h, x+h]} |f^{(3)}(x)|. \quad (4.1.6)$$

Таким образом, центральная разностная производная аппроксимирует производную $f'(x)$ со вторым порядком точности относительно параметра h .

Для вычисления первой производной можно получить и еще более сложные и точные формулы. Однако в таких формулах с ростом порядка точности возрастает и число используемых значений функции. Например,

$$f'(x) \approx \frac{f(x-2h) - 8f(x-h) + 8f(x+h) - f(x+2h)}{12h}, \text{ где } r(x, h) = O(h^4). \quad (4.1.7)$$

4.2. Формулы численного дифференцирования для второй производной

Наиболее простой и широко применяемой для приближенных вычислений второй производной является следующая формула

$$f''(x) \approx \frac{f(x-h) - 2f(x) + f(x+h)}{h^2}. \quad (4.2.1)$$

Она выводится из формулы $f''(x) \approx \frac{f'(x) - f'(x-h)}{h}$, в которой первые производные рассчитываются по формуле (4.2.1) по трем точкам $x-h$, x и $x+h$. Формулу (4.2.1) часто называют второй разностной производной. Покажем, что она имеет второй порядок точности относительно h . Итак, $r(x, h) = f''(x) - \frac{f(x-h) - 2f(x) + f(x+h)}{h^2}$, причем

$$f(x \pm h) = f(x) \pm f'(x)h + \frac{f''(x)}{2}h^2 \pm \frac{f'''(x)}{6}h^3 + \frac{f^{(4)}(\zeta_{\pm})}{24}h^4. \text{ Тогда}$$

$$r(x, h) = f''(x) - \frac{f(x) - f'(x)h + \frac{f''(x)}{2}h^2 - \frac{f^{(3)}(x)}{6}h^3 + \frac{f^{(4)}(\zeta_-)}{24}h^4 - 2f(x) + f(x)}{h^2} +$$

$$+\frac{f'(x)h + \frac{f''(x)}{2}h^2 + \frac{f^{(3)}(x)}{6}h^3 + \frac{f^{(4)}(\zeta_+)}{24}h^4}{h^2} = f''(x) - \frac{f''(x)h^2 + \frac{h^4}{24}(f^{(4)}(\zeta_-) + f^{(4)}(\zeta_+))}{h^2} = \\ = -\frac{h^4}{24}(f^{(4)}(\zeta_-) + f^{(4)}(\zeta_+)).$$

Следовательно, $|r(x, h)| \leq \frac{M_4}{12}h^2$, где $M_4 = \max_{[x-h, x+h]} |f^{(4)}(x)|$. (4.2.2)

Для получения $f''(x)$ можно использовать формулы любого порядка точности. Например, формула $f''(x) \approx \frac{-f(x-2h) + 16f(x-h) - 30f(x) + 16f(x+h) - f(x+2h)}{12h^2}$ (4.2.3)

имеет четвертый порядок точности относительно параметра h , но требует наличия значений функции в пяти точках.

4.3. Формулы численного дифференцирования, основанные на интерполяции алгебраическими многочленами

Предположим, что в окрестности точки x функция $y = f(x)$ аппроксимируется некоторой другой функцией $g(x)$, причем $g(x)$ в точке x легко вычисляется. Естественно в такой ситуации попытаться воспользоваться приближенной формулой $f^{(k)}(x) \approx g^{(k)}(x)$.

Пусть $P_n(x)$ - интерполяционный многочлен степени n с узлами интерполяции $x_0 < x_1 < \dots < x_n$, $x \in [x_0, x_n]$. В этом случае $f^{(k)}(x) \approx P_n^{(k)}(x)$, $0 \leq k \leq n$. Поскольку

$$P_n(x) = f(x_0) + f(x_0; x_1)(x - x_0) + f(x_0; x_1; x_2)(x - x_0)(x - x_1) + \dots + f(x_0; x_1; \dots; x_n)(x - x_0) \times \\ \times (x - x_1) \dots (x - x_{n-1}) = \sum_{k=0}^n f(x_0; x_1; \dots; x_k) \omega_k(x),$$

то для аппроксимации производных в общем случае при наличии неравномерной сетки узлов можно воспользоваться связью производных и разделенных разностей:

$$f^{(n)}(x) \approx n! f(x_0; x_1; \dots; x_n) = P_n^{(n)}(x). \quad (4.3.1)$$

Формула (4.3.1) имеет по крайней мере первый порядок точности. В частности при $n=1$ $f'(x) \approx f(x_0; x_1) = \frac{f(x_1) - f(x_0)}{x_1 - x_0}$ - первая разностная производная; при $n=2$

$$f''(x) \approx 2f(x_0; x_1; x_2) = \frac{2}{x_2 - x_0} \left[\frac{f(x_2) - f(x_1)}{x_2 - x_1} - \frac{f(x_1) - f(x_0)}{x_1 - x_0} \right] = \begin{cases} x_0 = x - h, x_1 = x, \\ x_2 = x + h, x_2 - x_0 = 2h, \\ x_2 - x_1 = h, x_1 - x_0 = h. \end{cases} = \\ = \frac{f(x-h) - 2f(x) + f(x+h)}{h^2} \text{ - вторая разностная производная.}$$

Если шаг сетки узлов постоянен, то можно вместо разделенных разностей использовать конечные:

$$f^{(n)}(x) \approx \frac{\Delta^n y_0}{h^n}. \quad (4.3.2)$$

4.4. Обусловленность формул численного дифференцирования

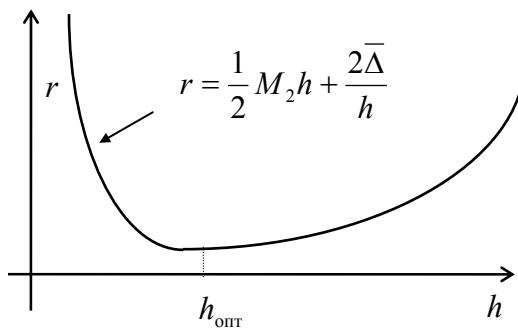
Несмотря на внешнюю простоту формул численного дифференцирования, их применение требует особой осторожности. Так как табличные значения $f^*(x)$ функции $y = f(x)$ непременно содержат ошибки, и эти ошибки являются неустранимыми, они прибавляются к погрешностям аппроксимации. Для уменьшения этой погрешности обычно уменьшают шаг h , но именно при малых шагах формулы численного дифференцирования становятся плохо

обусловленными и результат их применения может быть полностью искажен неустранимой ошибкой. Важно понимать, что действительная причина этого явления лежит не в несовершенстве методов вычисления производных, а в некорректности самой операции дифференцирования приближенно заданных функций.

Рассмотрим $r^*(x, h) = f'(x) - \frac{f^*(x+h) - f^*(x)}{h}$. Это полная погрешность, она складывается из погрешности аппроксимации $r_+(x, h) = f'(x) - \frac{f(x+h) - f(x)}{h}$ и неустранимой погрешности $r_H(x, h) = \frac{1}{h} [(f(x+h) - f^*(x+h)) - (f(x) - f^*(x))]$. Пусть $\bar{\Delta} = \max |\Delta(f^*(x))| = \max |f(x) - f^*(x)|$. Тогда $r_H(x, h)$ можно оценить следующим образом: $|r_H| \leq \frac{2\bar{\Delta}}{h}$. Фактически это число будет числом обусловленности формулы (4.1.1), то есть

$$\vartheta_\Delta = 2\bar{\Delta}/h. \quad (4.4.1)$$

Видно, что при $h \rightarrow 0$ $\vartheta_\Delta \rightarrow \infty$. Поэтому, несмотря на то, что погрешность аппроксимации стремится к нулю при $h \rightarrow 0$, полная погрешность $r(x, h) = r_+(x, h) + r_H = \frac{1}{2} M_2 h + \frac{2\bar{\Delta}}{h}$



будет неограниченно возрастать. Найдем $h_{\text{опт}}$, при котором $r(x, h) \rightarrow \min$. Для этого необходимо, чтобы $r'_h(x, h) = \frac{1}{2} M_2 - \frac{2\bar{\Delta}}{h^2} = 0$. Отсюда

$$h_{\text{опт}} = 2\sqrt{\frac{\bar{\Delta}}{M_2}}. \quad (4.4.2)$$

$$\text{Тогда } r_{\min} = r(h_{\text{опт}}) = 2\sqrt{\bar{\Delta} M_2}. \quad (4.4.3)$$

Таким образом, при использовании формул численного дифференцирования необходимо обращать внимание на выбор шага h . Даже при оптимальном выборе шага полная погрешность является величиной, пропорциональной $\sqrt{\bar{\Delta}}$. При $k > 1$ формулы для вычисления $f^{(k)}(x)$ обладают еще большей чувствительностью к ошибкам задания функций. Поэтому значения производных высокого порядка, найденные с помощью таких формул, могут быть очень неточными.

Пример. Пусть $f(x) = e^x$ задана на $[0, 1]$ с шагом $h = 0.2$ следующей таблицей:

x	0.0	0.2	0.4	0.6	0.8	1.0
$f(x)$	1.00000	1.22140	1.49182	1.82212	2.22554	2.71828

Найдем $f'(x)$ в узлах таблицы и оценим точность полученных данных. В точках $x = 0.0$ и $x = 1.0$ возможно применение только формул для левой и правой разностной производной:

$$f'(x_0) = f'(0.0) = \frac{f(0.2) - f(0.0)}{0.2} = \frac{1.22140 - 1.00000}{0.2} = 1.10700,$$

$$f'(x_n) = f'(1.0) = \frac{f(1.0) - f(0.8)}{0.2} = \frac{2.71828 - 2.22554}{0.2} = 2.46370.$$

В остальных точках применим формулу (4.1.5), имеющую более высокий порядок точности:

$$f'(0.2) = \frac{f(0.4) - f(0.0)}{0.4} = \frac{1.49182 - 1.00000}{0.4} = 1.22955,$$

$$f'(0.4) = \frac{f(0.6) - f(0.2)}{0.4} = \frac{1.82212 - 1.22140}{0.4} = 1.50180,$$

$$f'(0.6) = \frac{f(0.8) - f(0.4)}{0.4} = \frac{2.22554 - 1.49182}{0.4} = 1.83430,$$

$$f'(0.8) = \frac{f(1.0) - f(0.6)}{0.4} = \frac{2.71828 - 1.82212}{0.4} = 2.24040.$$

Сведем значения производной в таблицу, подобную исходной таблице задания функции:

x	0.0	0.2	0.4	0.6	0.8	1.0
$f'(x)$	1.10700	1.22955	1.50180	1.83430	2.24040	2.46370
$r(x)$	-0.10700	-0.00815	-0.00998	-0.01218	-0.01486	0.25458

$r(x)$ в данном случае легко вычисляется, так как $f'(x) = e^x = f(x)$. Погрешности можно было бы вычислить и по приведенным формулам для $r(x, h)$. Например, в точке

$$x = 0.0 \quad |r_{x=0}| \leq \frac{1}{2} M_2 h = \frac{1}{2} \max_{[0, 0.2]} |e^x| h = \frac{1}{2} e^{0.2} \cdot 0.2 = 0.5 \cdot 0.2 \cdot 1.22140 = 0.12214.$$

4.5. Простейшие квадратурные методы численного интегрирования

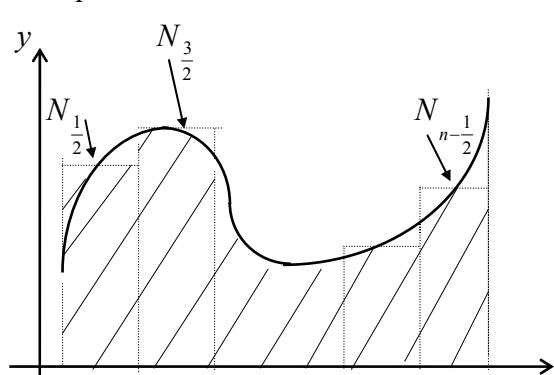
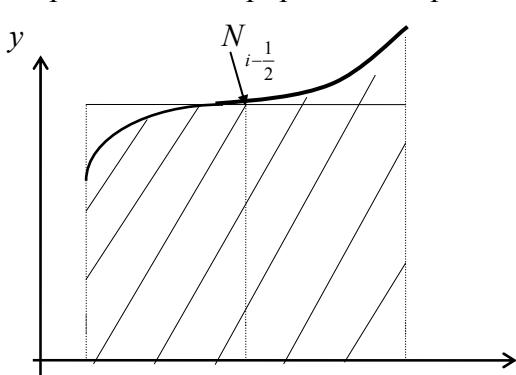
В прикладных исследованиях, когда возникает необходимость вычисления $\int_a^b f(x) dx$

и первообразной не существует, приходиться интеграл считать численно. Наиболее широко на практике используются квадратурные формулы - приближенные равенства вида

$$\int_a^b f(x) dx \approx \sum_{i=0}^N A_i f(\tilde{x}_i), \quad (4.5.1)$$

где \tilde{x}_i - некоторые точки из отрезка $[a, b]$ - узлы квадратурной формулы (4.5.1), A_i - числовые коэффициенты, называемые весами квадратурной формулы, $N \geq 0$ - целое число. Сумма $\sum_{i=0}^N A_i f(\tilde{x}_i)$, которая принимается за приближенное значение интеграла, называется квадратурной суммой.

Величина $R = \int_a^b f(x) dx - \sum_{i=0}^N A_i f(\tilde{x}_i)$ называется погрешностью или остаточным членом квадратурной формулы. Выведем простейшие квадратурные формулы, исходя из геометрической интерпретации определенного интеграла:



$$x_{i-1} \quad x_{\frac{i-1}{2}} \quad x_i$$

$$a = x_0 \quad x_1 \quad x_2 \quad \dots \quad x_{n-2} \quad x_{n-1} \quad x_n = b$$

Будем интерпретировать интеграл $I = \int_a^b f(x)dx$ как площадь криволинейной трапеции, ограниченной графиком функции $y = f(x)$ ($f(x) > 0$), осью абсцисс и прямыми $x = a$ и $x = b$. Разобьем отрезок $[a, b]$ на элементарные отрезки $[x_{i-1}, x_i]$ точками $a = x_0 < x_1 < \dots < x_n = b$. Интеграл I представится таким образом:

$$I = \sum_{i=1}^n I_i, \quad I_i = \int_{x_{i-1}}^{x_i} f(x)dx. \quad (4.5.2)$$

Введем обозначения: $f_i = f(x_i)$, $f_{\frac{i-1}{2}} = f\left(x_{\frac{i-1}{2}}\right)$, где $x_{\frac{i-1}{2}} = \frac{x_{i-1} + x_i}{2}$, $h = \text{const}$.

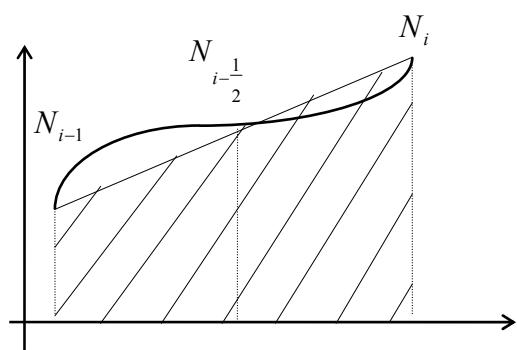
Формула центральных прямоугольников. Заменим приближенно площадь элементарной криволинейной трапеции площадью прямоугольника, основанием которого является отрезок $[x_{i-1}, x_i]$, а высота равна значению $f_{\frac{i-1}{2}}$. Тогда сразу получается элементарная квадратурная формула прямоугольников $I_i \approx h f_{\frac{i-1}{2}}$. Суммируя I_i по всему отрезку $[a, b]$, получим

$$I \approx I_{\text{прям}}^h = h \left(f_{\frac{1}{2}} + f_{\frac{3}{2}} + f_{\frac{5}{2}} + \dots + f_{\frac{n-1}{2}} \right) = h \sum_{i=1}^n f_{\frac{i-1}{2}}. \quad (4.5.3)$$

Совершенно аналогично можно получить формулы $I \approx h \sum_{i=0}^{n-1} f_i$ и $I \approx h \sum_{i=1}^n f_i$, которые называются квадратурными формулами левых и правых прямоугольников. Их точность $O(h)$, тогда как точность формулы (4.5.3) $O(h^2)$.

Формула трапеций. Соединив точки $N_{i-1}(x_{i-1}, f_{i-1})$ и $N_i(x_i, f_i)$, получим формулу трапеций. Заменим площадь элементарной криволинейной трапеции площадью построенной фигуры. Тогда $I_i \approx \frac{h}{2}(f_{i-1} + f_i)$, а итоговая формула примет вид

$$\begin{aligned} I \approx I_{\text{трап}}^h &= h \left(\frac{f_0}{2} + f_1 + f_2 + \dots + f_{n-1} + \frac{f_n}{2} \right) = \\ &= h \left(\frac{f_0 + f_n}{2} + \sum_{i=1}^{n-1} f_i \right). \end{aligned} \quad (4.5.4)$$

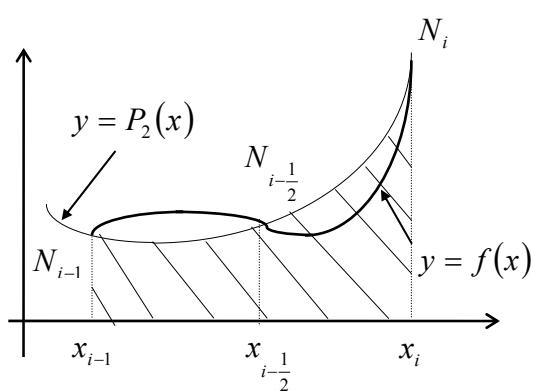


Заменим площадь элементарной криволинейной трапеции площадью построенной фигуры. Тогда $I_i \approx \frac{h}{2}(f_{i-1} + f_i)$, а итоговая формула примет вид

$$\begin{aligned} I \approx I_{\text{трап}}^h &= h \left(\frac{f_0}{2} + f_1 + f_2 + \dots + f_{n-1} + \frac{f_n}{2} \right) = \\ &= h \left(\frac{f_0 + f_n}{2} + \sum_{i=1}^{n-1} f_i \right). \end{aligned} \quad (4.5.4)$$

Формула парабол (Симпсона*). Если площадь элементарной криволинейной трапеции заменить площадью фигуры, расположенной под параболой, проходящей через точки $N_{i-1}, N_{i-1/2}, N_i$, то получим приближенное равенство $I_i \approx \int_{x_{i-1}}^{x_i} P_2(x)dx$, где $P_2(x)$ - интерполяционный многочлен второй степени с узлами $x_{i-1}, x_{i-1/2}, x_i$.

Для этих точек справедлива формула



* Томас Симпсон (1710-1761) - английский математик.

$$P_2(x) = f_{\frac{i-1}{2}} + \frac{f_i - f_{i-1}}{h} \left(x - x_{\frac{i-1}{2}} \right) + \frac{\frac{f_i - 2f_{\frac{i-1}{2}} + f_{i-1}}{h^2}}{2} \left(x - x_{\frac{i-1}{2}} \right)^2. \quad (4.5.5)$$

Действительно, легко проверить, что $f_{i-1} = P_2(x_{i-1})$, $f_{\frac{i-1}{2}} = P_2\left(x_{\frac{i-1}{2}}\right)$, $f_i = P_2(x_i)$. Кроме того, формула (4.5.5) относительно x представляет уравнение второй степени. Подставим теперь точки x_{i-1} , $x_{\frac{i-1}{2}}$, x_i в уравнение (4.5.5). Получим

$$\begin{aligned} P_2\left(x_{\frac{i-1}{2}}\right) &= f_{\frac{i-1}{2}}, \\ P_2(x_{i-1}) &= f_{\frac{i-1}{2}} + \frac{f_i - f_{i-1}}{h} \left(-\frac{h}{2} \right) + \frac{2\left(f_i - 2f_{\frac{i-1}{2}} + f_{i-1}\right)}{h^2} \frac{h^2}{4} = f_{\frac{i-1}{2}} - \frac{f_i}{2} + \frac{f_{i-1}}{2} + \frac{f_i}{2} - f_{\frac{i-1}{2}} + \frac{f_{i-1}}{2} = f_{i-1}, \\ P_2(x_i) &= f_{\frac{i-1}{2}} + \frac{f_i - f_{i-1}}{h} \frac{h}{2} + \frac{2\left(f_i - 2f_{\frac{i-1}{2}} + f_{i-1}\right)}{h^2} \frac{h^2}{4} = f_{\frac{i-1}{2}} + \frac{f_i}{2} - \frac{f_{i-1}}{2} + \frac{f_i}{2} - f_{\frac{i-1}{2}} + \frac{f_{i-1}}{2} = f_i. \end{aligned}$$

Таким образом, точки x_{i-1} , $x_{\frac{i-1}{2}}$, x_i параболы $P_2(x)$ удовлетворяют уравнению (4.5.5).

Проинтегрируем полученную формулу по отрезку $[x_{i-1}, x_i]$. Тогда

$$\begin{aligned} \int_{x_{i-1}}^{x_i} P_2(x) dx &= hf_{\frac{i-1}{2}} + \frac{f_i - f_{i-1}}{h} \int_{x_{i-1}}^{x_i} \left(x - x_{\frac{i-1}{2}} \right) dx + \frac{\frac{f_i - 2f_{\frac{i-1}{2}} + f_{i-1}}{h^2}}{2} \int_{x_{i-1}}^{x_i} \left(x - x_{\frac{i-1}{2}} \right)^2 dx = hf_{\frac{i-1}{2}} + \\ &+ \frac{f_i - f_{i-1}}{h} \frac{\left(x - x_{\frac{i-1}{2}} \right)^2 \Big|_{x_{i-1}}^{x_i}}{2} + \frac{f_i - 2f_{\frac{i-1}{2}} + f_{i-1}}{h^2} \frac{\left(x - x_{\frac{i-1}{2}} \right)^3 \Big|_{x_{i-1}}^{x_i}}{3} = hf_{\frac{i-1}{2}} + \frac{f_i - f_{i-1}}{h} \left(\frac{h^2}{8} - \frac{h^2}{8} \right) + \\ &+ \frac{f_i - 2f_{\frac{i-1}{2}} + f_{i-1}}{h^2} \left(\frac{h^3}{24} + \frac{h^3}{24} \right) = hf_{\frac{i-1}{2}} + \frac{h}{6} f_i - \frac{h}{3} f_{\frac{i-1}{2}} + \frac{h}{6} f_{i-1} = \frac{h}{6} \left(f_{i-1} + 4f_{\frac{i-1}{2}} + f_i \right). \end{aligned}$$

Применяя полученную формулу на каждом элементарном отрезке, получим квадратурную формулу парабол (Симпсона)

$$I \approx I_{\text{пара}}^h = \frac{h}{6} \left(f_0 + f_n + 4 \sum_{i=1}^n f_{\frac{i-1}{2}} + 2 \sum_{i=1}^{n-1} f_i \right). \quad (4.5.6)$$

4.6. Оценка погрешностей простейших квадратурных формул

Теорема 4.1. Пусть функция $y = f(x)$ дважды непрерывно дифференцируема на $[a, b]$. Тогда для формул (4.5.3) и (4.5.4) справедливы следующие оценки погрешности:

$$\left| I - I_{\text{прям}}^h \right| \leq \frac{M_2}{24} (b-a) h^2, \quad (4.6.1)$$

$$\left| I - I_{\text{трап}}^h \right| \leq \frac{M_2}{12} (b-a) h^2. \quad (4.6.2)$$

Доказательство

Получим, например, формулу (4.6.1). По определению погрешности

$$R = \int_a^b f(x) dx - h \sum_{i=1}^n f_{i-\frac{1}{2}} = \sum_{i=1}^n \int_{x_{i-1}}^{x_i} \left[f(x) - f\left(x_{i-\frac{1}{2}}\right) \right] dx.$$

С другой стороны, по формуле Тейлора

$$f(x) = f\left(x_{i-\frac{1}{2}}\right) + f'\left(x_{i-\frac{1}{2}}\right) \left(x - x_{i-\frac{1}{2}} \right) + \frac{f''(\zeta)}{2} \left(x - x_{i-\frac{1}{2}} \right)^2, \text{ где } x \in [x_{i-1}, x_i], \zeta \in (x_{i-1}, x_i).$$

$$\begin{aligned} \text{Тогда } R_i &= \int_{x_{i-1}}^{x_i} \left[f(x) - f\left(x_{i-\frac{1}{2}}\right) \right] dx = \int_{x_{i-1}}^{x_i} \left[f'\left(x_{i-\frac{1}{2}}\right) \left(x - x_{i-\frac{1}{2}} \right) + \frac{f''(\zeta)}{2} \left(x - x_{i-\frac{1}{2}} \right)^2 \right] dx = \\ &= f'\left(x_{i-\frac{1}{2}}\right) \int_{x_{i-1}}^{x_i} \left(x - x_{i-\frac{1}{2}} \right) dx + \frac{f''(\zeta)}{2} \int_{x_{i-1}}^{x_i} \left(x - x_{i-\frac{1}{2}} \right)^2 dx \leq \frac{M_2}{2} \int_{x_{i-1}}^{x_i} \left(x - x_{i-\frac{1}{2}} \right)^2 dx = \\ &= \frac{M_2}{6} \left(x - x_{i-\frac{1}{2}} \right)^3 \Big|_{x_{i-1}}^{x_i} = \frac{M_2}{6} \left(\frac{h^3}{8} + \frac{h^3}{8} \right) = \frac{M_2}{24} h^3. \end{aligned}$$

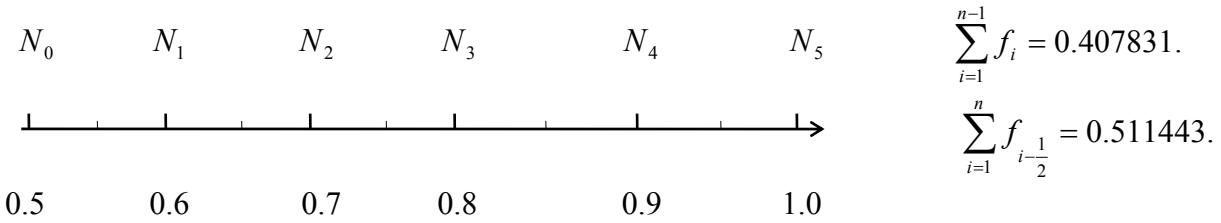
Но $(b-a) = nh$, следовательно, $R = R_i n \leq \frac{M_2}{24} h^2 n \frac{b-a}{n} = \frac{M_2(b-a)}{24} h^2$. Аналогично выводится и формула (4.6.2).

Теорема 4.2. Пусть функция $y = f(x)$ имеет на $[a, b]$ непрерывную четвертую производную. Тогда для формулы (4.5.6) справедлива следующая оценка:

$$\left| I - I_{\text{параб}}^h \right| \leq \frac{M_4(b-a)}{2880} h^4. \quad (4.6.3)$$

Пример. Вычислить интеграл $\int_{0.5}^1 \frac{\cos(0.4x^2 + 1)}{2.3 + \sin(1.5x + 0.3)} dx$ по формулам прямоугольников, трапеций и парабол и оценить погрешность вычислений.

i	x_i	$0.4x^2 + 1$	числитель	$1.5x + 0.3$	знаменат.	y_i
0	0.50	1.100	0.453596	1.050	3.167423	0.143207
1/2	0.55	1.121	0.434782	1.125	3.202268	0.135773
1	0.60	1.144	0.413957	1.200	3.232039	0.128079
3/2	0.65	1.169	0.391072	1.275	3.256570	0.120088
2	0.70	1.196	0.366083	1.350	3.275723	0.111756
5/2	0.75	1.225	0.338946	1.425	3.289391	0.103042
3	0.80	1.256	0.309623	1.500	3.297495	0.093896
7/2	0.85	1.272	0.294370	1.575	3.299991	0.089203
4	0.90	1.324	0.244299	1.650	3.296865	0.074100
9/2	0.95	1.361	0.208261	1.725	3.288134	0.063337
5	1.00	1.400	0.169967	1.800	3.273848	0.051917



$$I_{\text{прям}}^h = h \sum_{i=1}^n f_{i-\frac{1}{2}} = 0.025572,$$

$$I_{\text{трап}}^h = h \left(\frac{f_0 + f_n}{2} + \sum_{i=1}^{n-1} f_i \right) = 0.05(0.097562 + 0.407831) = 0.025270,$$

$$I_{\text{параб}}^h = \frac{h}{6} \left(f_0 + f_n + 4 \sum_{i=1}^n f_{i-\frac{1}{2}} + 2 \sum_{i=1}^{n-1} f_i \right) = \frac{0.05}{6} (0.195124 + 2.045772 + 0.815662) = 0.025471.$$

Оценим теперь погрешность вычислений только по теореме 4.1:

$$f(x) = \frac{\cos(0.4x^2 + 1)}{2.3 + \sin(1.5x + 0.3)}, \quad f'(x) = \frac{-0.8x \sin(0.4x^2 + 1)(2.3 + \sin(1.5x + 0.3))}{(2.3 + \sin(1.5x + 0.3))^2} + \frac{-1.5 \cos(0.4x^2 + 1)\cos(1.5x + 0.3)}{(2.3 + \sin(1.5x + 0.3))^2}.$$

Вторая производная уже получается настолько громоздкой, что оценка погрешности становится затруднительной. Тем не менее можно провести грубую оценку сверху $f''(x) \leq 0.118 + 2.618x - 0.640x^2 \leq 2.000$. Итак, $|I - I_{\text{прям, трап}}^h| \leq \frac{2}{24} 0.5 \cdot 0.05^2 = 0.000104$.

Так оно и есть, поскольку значения интеграла, полученные по формулам прямоугольников и трапеций, отличаются в четвертом знаке после запятой.

4.7. Квадратурные формулы интерполяционного типа

Для приближенного вычисления определенных интегралов часто используют следующий прием. Подынтегральную функцию $f(x)$ аппроксимируют на $[a, b]$ функцией $g(x)$,

интеграл от которой легко вычисляется и полагают $\int_a^b f(x)dx \approx \int_a^b g(x)dx$. Еще чаще используется другой подход. Интеграл I представляют в виде суммы интегралов по отрезкам $[x_{i-1}, x_i]$.

На каждом таком отрезке $f(x)$ аппроксимируют некоторой легко интегрируемой функцией

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

Пусть t_0, t_1, \dots, t_m - сетка на отрезке $[-1, 1]$. Аппроксимируем $f(x)$ на i -м элементарном отрезке $[x_{i-1}, x_i]$ интерполяционным многочленом $P_{m,i}(x)$ с узлами

$z_j^{(i)} = x_{i-\frac{1}{2}} + \frac{t_j h_i}{2}$, $j = 0, 1, 2, \dots, m$. Например, можно воспользоваться многочленом Лагранжа

$$P_{m,i}(x) = \sum_{j=0}^m f(z_j^{(i)}) \varphi_{m,j}^{(i)}(x), \quad \varphi_{m,j}^{(i)}(x) = \prod_{k=0, k \neq j}^m \frac{x - z_k^{(i)}}{z_j^{(k)} - z_k^{(i)}}. \quad \text{Вычислим } \int_{x_{i-1}}^{x_i} P_{m,i}(x)dx =$$

$$= \int_{x_{i-1}}^{x_i} \sum_{j=0}^m f(z_j^{(i)}) \varphi_{m,j}^{(i)}(x) dx = \left\{ \begin{array}{l} x = x_{i-\frac{1}{2}} + \frac{th_i}{2}, \\ dx = \frac{h_i}{2} dt, \\ x = x_{i-1}, t = -1, x = x_i, t = 1. \end{array} \right\} = h_i \sum_{j=0}^m a_j f\left(x_{i-\frac{1}{2}} + t_j y h_i \frac{1}{2}\right), \text{ где} \\ a_j = \frac{1}{2} \prod_{\substack{1 \leq k \leq m \\ k \neq j}} \frac{t_j - t_k}{t_j - t_k} dt. \text{ Тогда } I^h \approx \sum_{i=1}^n \int_{x_{i-1}}^{x_i} P_{m,i}(x) dx = \sum_{i=1}^n h_i \sum_{j=0}^m a_j f\left(x_{i-\frac{1}{2}} + \frac{t_j h_i}{2}\right). \quad (4.7.1)$$

Формулы такого вида называются квадратурными формулами Ньютона - Котеса*.

4.8. Квадратурные формулы Гаусса

Итак, все квадратурные формулы имеют вид $\int_a^b f(x) dx \approx \sum_{i=0}^N A_i f(x_i)$, A_i - веса. Формулы Гаусса строятся также по общему виду с дополнительным условием: формула должна быть точной для многочленов наиболее высокой степени. Как правило, формулы Гаусса сначала строятся для отрезка $[-1, 1]$, то есть $\int_{-1}^1 f(x) dx \approx \sum_{i=0}^N a_i f(t_i)$, а затем с помощью замены $x = \frac{a+b}{2} + \frac{b-a}{2} t$ осуществляется переход к формулам интегрирования на произвольном отрезке: $\int_a^b f(x) dx \approx \frac{b-a}{2} \sum_{i=0}^N a_i f\left(\frac{a+b}{2} + \frac{b-a}{2} t_i\right)$.

Требование точности формулы для многочленов степени m эквивалентно требованию ее точности для функции $f(t): 1, t, t^2, \dots, t^m$ (базис для $P_m(x)$). Следовательно,

$$\sum_{i=0}^N a_i t_i^k = \int_{-1}^1 t^k dt = \frac{1 - (-1)^{k+1}}{k+1}, \quad k = 0, 1, \dots, m. \quad (4.8.1)$$

Уравнение (4.8.1) дает систему нелинейных уравнений для определения a_i и t_i . Этих переменных $a_0, a_1, \dots, a_N, t_0, t_1, \dots, t_N$ $2N$ штук. Следовательно, необходимый многочлен будет иметь степень $m = 2N + 1$, так как нужно учесть, что $t^0 = 1$. Построим, например, квадратурную формулу Гаусса с двумя и тремя узлами.

а). $N = 1, m = 2N + 1 = 3$. Индекс i пробегает значения $0, N$, то есть $i = 0, i = 1$ - два узла. По формуле (4.8.1) получим

$$\begin{aligned} k = 0, \quad a_0 + a_1 &= \int_{-1}^1 1 dt = 2, \\ k = 1, \quad a_0 t_0 + a_1 t_1 &= \int_{-1}^1 t dt = 0, \\ k = 2, \quad a_0 t_0^2 + a_1 t_1^2 &= \int_{-1}^1 t^2 dt = \frac{2}{3}, \\ k = 3, \quad a_0 t_0^3 + a_1 t_1^3 &= \int_{-1}^1 t^3 dt = 0. \end{aligned}$$

Итак, соответствующая нелинейная система уравнений имеет вид

* Роджер Котес (1682-1716) - английский математик.

$$\begin{cases} a_0 + a_1 = 2, \\ a_0 t_0 + a_1 t_1 = 0, \\ a_0 t_0^2 + a_1 t_1^2 = \frac{2}{3}, \\ a_0 t_0^3 + a_1 t_1^3 = 0. \end{cases} \Rightarrow \begin{cases} a_0 = a_1 = 1, \\ t_0 = -\frac{1}{\sqrt{3}}, t_1 = \frac{1}{\sqrt{3}}. \end{cases}$$

Таким образом, получаем квадратурную формулу Гаусса:

$$\int_{-1}^1 f(t) dt \approx f\left(-\frac{1}{\sqrt{3}}\right) + f\left(\frac{1}{\sqrt{3}}\right), \quad (4.8.2)$$

точную для многочленов третьей степени.

б). $N = 2, m = 2N + 1 = 5, i = 0, 1, 2$. Формула (4.8.1) дает

$$\begin{aligned} k = 0, \quad a_0 + a_1 + a_2 &= \int_{-1}^1 1 dt = 2, \\ k = 1, \quad a_0 t_0 + a_1 t_1 + a_2 t_2 &= \int_{-1}^1 t dt = 0, \\ k = 2, \quad a_0 t_0^2 + a_1 t_1^2 + a_2 t_2^2 &= \int_{-1}^1 t^2 dt = \frac{2}{3}, \\ k = 3, \quad a_0 t_0^3 + a_1 t_1^3 + a_2 t_2^3 &= \int_{-1}^1 t^3 dt = 0, \\ k = 4, \quad a_0 t_0^4 + a_1 t_1^4 + a_2 t_2^4 &= \int_{-1}^1 t^4 dt = \frac{2}{5}, \\ k = 5, \quad a_0 t_0^5 + a_1 t_1^5 + a_2 t_2^5 &= \int_{-1}^1 t^5 dt = 0. \end{aligned}$$

Аналогично для нахождения переменных a_j и t_j будем иметь систему уравнений

$$\begin{cases} a_0 + a_1 + a_2 = 2, \\ a_0 t_0 + a_1 t_1 + a_2 t_2 = 0, \\ a_0 t_0^2 + a_1 t_1^2 + a_2 t_2^2 = \frac{2}{3}, \\ a_0 t_0^3 + a_1 t_1^3 + a_2 t_2^3 = 0, \\ a_0 t_0^4 + a_1 t_1^4 + a_2 t_2^4 = \frac{2}{5}, \\ a_0 t_0^5 + a_1 t_1^5 + a_2 t_2^5 = 0. \end{cases} \Rightarrow \begin{cases} a_0 = 0, a_1 = a_2 = a_3 = \frac{8}{9}, \\ t_0 = 0, t_1 = -\frac{1}{\sqrt{\frac{5}{3}}} = -0.774576, t_2 = 0, t_3 = \frac{1}{\sqrt{\frac{5}{3}}} = 0.774576. \end{cases}$$

Можно показать, что значения t_j являются корнями многочленов Лежандра* $P_{N+1}(t)$, а

$$a_j = \frac{2}{(1-t_j^2)[P_{N+1}'(t_j)]^2}, \text{ где } P_0(t)=1, P_1(t)=t, P_2(t)=\frac{3}{2}\left(t^2-\frac{1}{3}\right) \text{ и так далее.}$$

Итак, квадратурная формула Гаусса в случае б) имеет вид

$$\int_{-1}^1 f(t) dt \approx \frac{8}{9} f\left(-\frac{1}{\sqrt{\frac{5}{3}}}\right) + \frac{8}{9} f(0) + \frac{8}{9} f\left(\frac{1}{\sqrt{\frac{5}{3}}}\right). \quad (4.8.3)$$

Она точна до многочленов пятой степени. На таких же принципах могут быть построены квадратурные формулы Чебышева, Лагерра**, Эрмита.

Для квадратурной формулы Гаусса справедлива следующая оценка погрешности:

* Андриен Мари Лежандр (1752-1833) - французский математик.

** Эдмунд Николя Лагерр (1834-1886) - французский математик.

$$|R| \leq \alpha_N M_{2N+2} (b-a)^{2N+3},$$

$$\alpha_N = \frac{[(N+1)]^4}{(2N+3)[(2N+2)]^3}. \quad (4.8.4)$$

Ее коэффициенты убывают очень быстро, например, $\alpha_4 < 10^{-13}$.

4.9. Лабораторная работа № 8. Численное дифференцирование и численное интегрирование функций

Численное дифференцирование применяется в тех случаях, когда либо невозможно, либо очень сложно или дорогое продифференцировать функцию аналитически. Из-за быстрого накопления ошибок при численном вычислении старших производных обычно ограничиваются нахождением первой и второй производной по формулам (4.1.1), (4.1.2) и (4.1.5).

Одним из способов повышения точности вычислений является уменьшение шага до предела, определяемого выражением (4.4.1). Однако если функция задана на сетке, то при численном дифференцировании невозможно выбрать шаг h , меньший шага сетки, и тогда формулы (4.1.1)-(4.1.5) могут давать слишком большую погрешность. В этом случае исходную функцию часто аппроксимируют какой-либо гладкой функцией, значение производной от которой принимают за приближенное значение искомой производной.

Пример. Пусть дана функция $y = \frac{x}{2} \cdot (\sin \ln x + \cos \ln x)$. Найдем ее первую и вторую

производные, вычислим правую, левую и центральную первые разностные производные, вторую разностную производную по формуле (4.2.1) на отрезке $x \in [0.5, 1.5]$ и исследуем влияние шага h на точность вычислений.

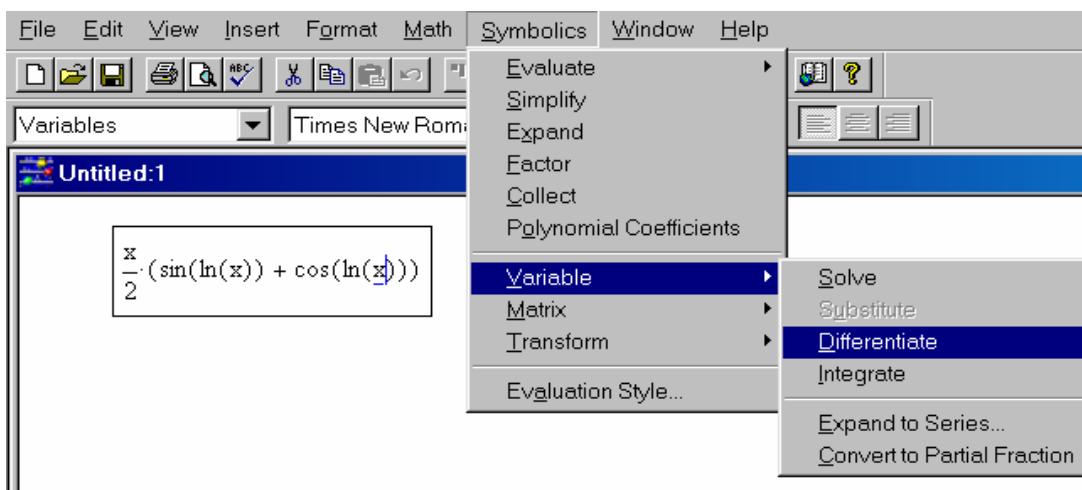
Легко проверить, что $y' = \cos \ln x$, $y'' = -\frac{1}{x} \cdot \sin \ln x$. Выберем первоначально $h = 0.2$.

Введем с клавиатуры исходные данные и построим графики $y(x)$, $y'(x)$ и $y''(x)$:

$$\text{ORIGIN := 1 } f(x) := \frac{x}{2} \cdot (\sin(\ln(x)) + \cos(\ln(x))).$$

Первая и вторая производные этой функции уже найдены. Однако это можно было сделать в среде пакета Mathcad, причем двумя способами: с помощью панели инструментов Calculus (Панель операций математического анализа) и через меню символьных операций Symbolics.

С помощью меню производная находится следующим образом. В рабочий документ надо ввести выражение дифференцируемой функции, выделить синим уголком аргумент (переменную дифференцирования) и щелкнуть по строке Differentiate в пункте Variable меню Symbolics:



Ниже появится выражение для производной:

$$\frac{1}{2} \cdot \sin(\ln(x)) + \frac{1}{2} \cdot \cos(\ln(x)) + \frac{1}{2} \cdot x \cdot \left(\frac{\cos(\ln(x))}{x} - \frac{\sin(\ln(x))}{x} \right).$$

Выделяем все выражение и выбираем пункт Simplify в меню Symbolics. Ниже получим $\cos(\ln(x))$.

Вторая возможность связана с оператором символьных преобразований и оператором символьных преобразований с ключевым словом панели ключевых слов символьных выражений. Набираем оператор . В квадратик вставляем оператор $\frac{df(x)}{dx}$ из панели Calculus. После щелчка вне этого оператора слева от него появляется выражение для производной, которое можно упростить, выбрав оператор simplify в той же панели ключевых слов символьных выражений. В итоге получим следующее выражение:

$$\frac{df(x)}{dx} \rightarrow \frac{1}{2} \cdot \sin(\ln(x)) + \frac{1}{2} \cdot \cos(\ln(x)) + \frac{1}{2} \cdot x \cdot \left(\frac{\cos(\ln(x))}{x} - \frac{\sin(\ln(x))}{x} \right) \text{simplify} \rightarrow \cos(\ln(x)).$$

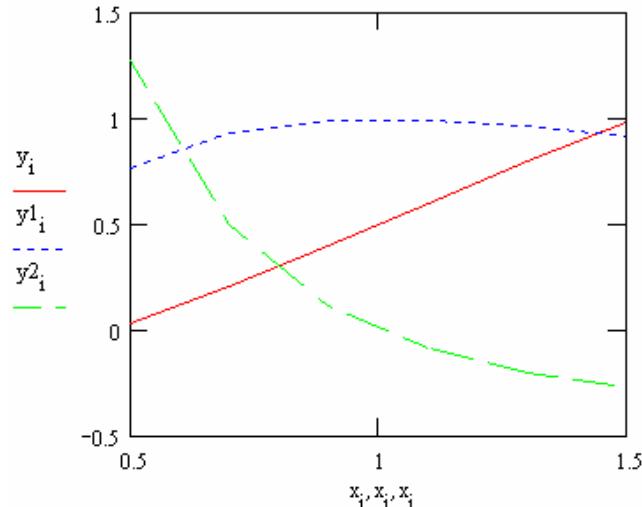
Аналогичные действия нужно выполнить для нахождения второй производной, которая сразу получается в окончательном виде и более не упрощается оператором simplify.

Вводим с клавиатуры следующую информацию:

$$f1(x) := \cos(\ln(x)) \quad f2(x) := \frac{-1}{x} \cdot \sin(\ln(x)) \quad a := 0.5 \quad b := 1.5 \quad h := 0.2 \quad n := \frac{(b-a)}{h}$$

$$i := 1..n+1 \quad x_i := a + h \cdot (i-1) \quad y_i := f(x_i) \quad y1_i := f1(x_i) \quad y2_i := f2(x_i).$$

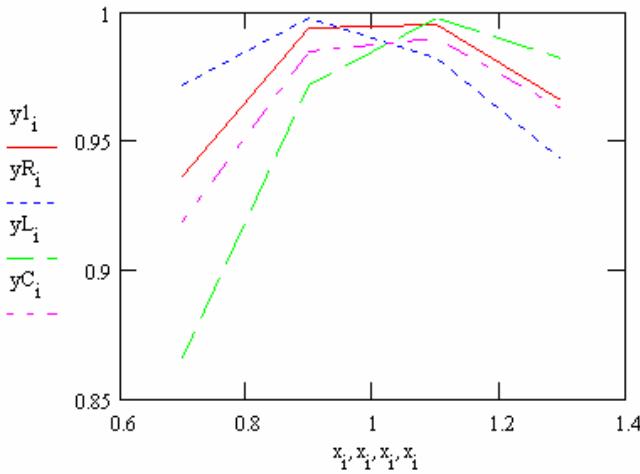
В точке $x = a$ возможно вычисление только правой разностной производной по формуле (4.1.1), в точке $x = b$ - только левой. Центральная разностная производная по формуле (4.1.5) не может быть посчитана при $x = a$ и $x = b$. Вводим следующую часть программы:



$$i := 1..n \quad yR_i := \frac{(y_{i+1} - y_i)}{h} \quad yR_{n+1} := 0$$

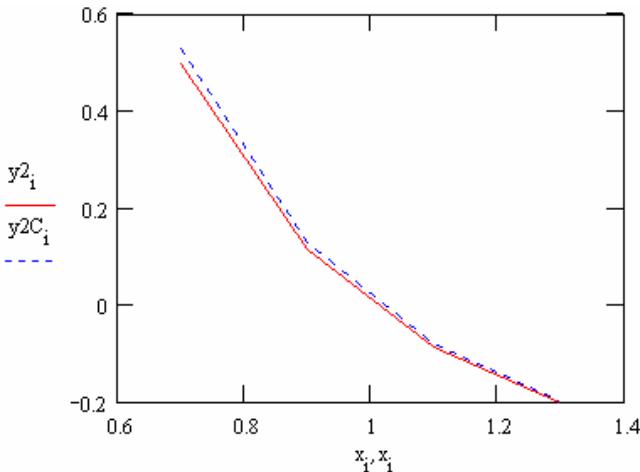
$$i := 2..n+1 \quad yL_i := \frac{(y_i - y_{i-1})}{h} \quad yR_1 := 0$$

$$i := 2..n \quad yC_i := \frac{(y_{i+1} - y_{i-1})}{2 \cdot h} \quad yC_1 := 0 \quad yC_{n+1} := 0$$

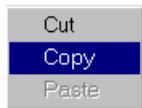


Наконец, находим вторую разностную производную по формуле (4.2.1). Она также не может быть посчитана в первой и последней точках заданной сетки узлов:

$$y2C_1 := 0 \quad y2C_{n+1} := 0 \quad i := 2 \dots n \quad y2C_i := \frac{(y_{i-1} - 2 \cdot y_i + y_{i+1})}{h^2}$$

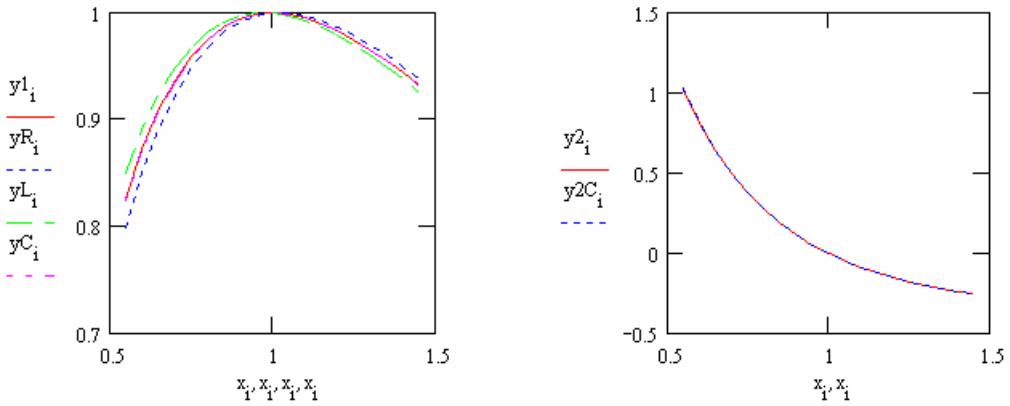


Уменьшим теперь шаг, например, в четыре раза. Для построения графиков необходимо пересчитать все значения функции и производных во всех новых точках сетки. Скопируем всю программу вместе с графиками после оператора $h := 0.005$. Для этого надо, нажав левую мышку и не отпуская ее, обвести копируемую часть пунктирной линией. Затем щелкнуть по любому месту обведенной части правой мышкой. Появится дополнительное меню



Необходимо выбрать пункт копирования (Copy). Обведенная пунктирной линией часть программы скопируется в буфер обмена. Затем необходимо подвести курсор к нужному месту окна рабочего документа. Щелкнув в этом месте еще раз правой мышкой, раскроем то же дополнительное меню и выберем на этот раз пункт вставки (Paste). Копируемый кусок программы поместится ниже курсора. Приведем здесь лишь графики первых и вторых производных.

Видно, что точность представления производных заметно возросла. На первом графике первая центральная разностная производная, а на втором графике вторая разностная производная практически совпали с точными значениями, вычисленными по аналитическим выражениям:



Задание № 1. Для данных функций построить правую, левую и центральную первые разностные производные, вторую разностную производную на указанном интервале с данными шагами сетки h_1 и h_2 и сравнить полученные значения с точными значениями производных:

1. $y = \ln(x^2 + \sqrt{x^4 + 1})$, $x \in [1, 2.5]$, $h_1 = 0.3$, $h_2 = 0.05$.
2. $y = 2^{\sin x^2}$, $x \in [0, 2]$, $h_1 = 0.4$, $h_2 = 0.05$.
3. $y = \sin \cos^2 x \cdot \cos \sin^2 x$, $x \in \left[0, \frac{\pi}{2}\right]$, $h_1 = \frac{\pi}{10}$, $h_2 = \frac{\pi}{50}$.
4. $y = \cos \frac{1}{\log_2 x}$, $x \in [1, 5]$, $h_1 = 0.5$, $h_2 = 0.05$.
5. $y = \frac{1}{\sqrt{2}} \operatorname{arctg} \frac{x^2 - 1}{\sqrt{2x}}$, $x \in [1, 2]$, $h_1 = 0.2$, $h_2 = 0.05$.
6. $y = e^{\sqrt{\frac{1-x}{1+x}}}$, $x \in [0, 1]$, $h_1 = 0.2$, $h_2 = 0.05$.
7. $y = \ln \ln \ln x^2$, $x \in [1, 10]$, $h_1 = 2$, $h_2 = 0.5$.
8. $y = \sqrt{2x^2 + \sqrt{x^2 + 1}}$, $x \in [-1, 1]$, $h_1 = 0.4$, $h_2 = 0.1$.
9. $y = e^{-\frac{x^2}{2}}$, $x \in [-2, 2]$, $h_1 = 0.4$, $h_2 = 0.1$.
10. $y = \arccos(\sin x^4 - \cos x^4)$, $x \in \left[0, \frac{\pi}{2}\right]$, $h_1 = \frac{\pi}{5}$, $h_2 = \frac{\pi}{20}$.
11. $y = \sin(\arcsin x)$, $x \in [0, 1]$, $h_1 = 0.2$, $h_2 = 0.05$.
12. $y = 3^{\operatorname{arcctg}(2x+\pi)}$, $x \in \left[-\frac{\pi}{2}, \frac{\pi}{2}\right]$, $h_1 = \frac{\pi}{5}$, $h_2 = \frac{\pi}{25}$.
13. $y = \ln \ln \left(\frac{x}{2}\right)$, $x \in [1, 10]$, $h_1 = 1$, $h_2 = 0.25$.
14. $y = 2^{\cos^2 x}$, $x \in [0, 2]$, $h_1 = 0.4$, $h_2 = 0.05$.
15. $y = \log_2^3(2x+3)^2$, $x \in [1, 8]$, $h_1 = 0.5$, $h_2 = 0.1$.
16. $y = \cos(3 \arccos x)$, $x \in [-1, 1]$, $h_1 = 0.4$, $h_2 = 0.1$.
17. $y = 10^{\frac{x}{\log_3 x}}$, $x \in [1, 9]$, $h_1 = 1$, $h_2 = 0.2$.

$$18. \ y = \operatorname{arcctg} 2^x, \ x \in [1, 3], \ h_1 = 0.5, \ h_2 = 0.1.$$

$$19. \ y = \cos \frac{1}{x}, \ x \in [1, 4], \ h_1 = 0.5, \ h_2 = 0.05.$$

$$20. \ y = \ln(x + \sqrt{1+x^2}), \ x \in [0, 1], \ h_1 = 0.2, \ h_2 = 0.05.$$

$$21. \ y = \sqrt{1-x^2} \cdot \arcsin x, \ x \in [-1, 1], \ h_1 = 0.25, \ h_2 = 0.05.$$

$$22. \ y = xe^{x^2}, \ x \in [0, 1], \ h_1 = 0.1, \ h_2 = 0.02.$$

$$23. \ y = x \ln x, \ x \in [1, 10], \ h_1 = 1, \ h_2 = 0.25.$$

$$24. \ y = \sqrt{4-x^2}, \ x \in [0, 2], \ h_1 = 0.25, \ h_2 = 0.05.$$

$$25. \ y = \frac{\arcsin x}{\sqrt{1-x^2}}, \ x \in [-0.9, 0.9], \ h_1 = 0.2, \ h_2 = 0.05.$$

$$26. \ y = \sin x \cdot e^{\cos x}, \ x \in \left[0, \frac{\pi}{2}\right], \ h_1 = \frac{\pi}{20}, \ h_2 = \frac{\pi}{50}.$$

$$27. \ y = \ln \frac{1-e^x}{e^x}, \ x \in [-3, 1], \ h_1 = 0.25, \ h_2 = 0.05.$$

$$28. \ y = \arccos \sqrt{1-3x}, \ x \in [-2, 0], \ h_1 = 0.25, \ h_2 = 0.05.$$

$$29. \ y = e^x (\sin 3x - 3 \cos 3x), \ x \in \left[0, \frac{\pi}{4}\right], \ h_1 = \frac{\pi}{40}, \ h_2 = \frac{\pi}{200}.$$

$$30. \ y = \operatorname{arctg} \frac{x+1}{x-1}, \ x \in [0, \pi], \ h_1 = \frac{\pi}{6}, \ h_2 = \frac{\pi}{30}.$$

В среде Mathcad возможно численное интегрирование заданных функций и символьное интегрирование. Символьное интегрирование можно выполнить двумя способами, так же как и дифференцирование: с помощью панели операций математического анализа и через меню символьных операций.

С помощью меню нужно выполнить аналогичные действия, что и при дифференцировании: ввести выражение интегрируемой функции, выделить аргумент и щелкнуть по строке Integrate в пункте Variable меню Symbolics. В случае $f(x)$ и $f1(x)$ получим

$$\begin{aligned} & \frac{x}{2} \cdot (\sin(\ln(x)) + \cos(\ln(x))) \\ & \frac{1}{20} x^{(2+i)} - \frac{3}{20} \cdot i \cdot x^{(2+i)} + \frac{1}{20} x^{(2-i)} - \frac{3}{20} \cdot i \cdot x^{(2-i)} \\ & \quad \cos(\ln(x)) \\ & \frac{x}{2} \cdot (\sin(\ln(x)) + \cos(\ln(x))). \end{aligned}$$

Таким образом, только интеграл от $f1(x)$ имеет действительное аналитическое выражение, $\int f(x)dx$ выражается в комплексном виде. Аналогичный результат получается при использовании оператора символьных преобразований

$$\begin{aligned} \int f(x)dx & \rightarrow \frac{1}{20} x^{(2+i)} - \frac{3}{20} \cdot i \cdot x^{(2+i)} + \frac{1}{20} x^{(2-i)} - \frac{3}{20} \cdot i \cdot x^{(2-i)} \\ \int f1(x)dx & \rightarrow \frac{1}{2} \cdot x \cdot (\sin(\ln(x)) + \cos(\ln(x))). \end{aligned}$$

Численное интегрирование также можно осуществить двумя путями: с помощью панели операций математического анализа  и панели равенств и отношений .

В среде Mathcad знак $\langle = \rangle$ означает числовой, а знак $\langle \rightarrow \rangle$ символьный вывод значения переменной, функции или выражения, записанного до этого знака. При численном интегрировании удобнее пользоваться оператором $\langle = \rangle$. В нашем случае

$$\begin{aligned} \int_{0.5}^{1.5} f(x)dx &\rightarrow .50169909119550428276 - 1 \cdot 10^{-21} \cdot i \\ \int_{0.5}^{1.5} f(x)dx &= 0.5016991 \\ \int_{0.5}^{1.5} f1(x)dx &\rightarrow .95245452085281105139 \\ \int_{0.5}^{1.5} f1(x)dx &= 0.9524545. \end{aligned}$$

Формулы численного интегрирования (4.5.3), (4.5.4) и (4.5.6) настолько просты, что значение интеграла можно подсчитать, используя несколько арифметических операторов или с помощью небольшой подпрограммы. Например, формула центральных прямоугольников реализуется таким образом:

$$i := 1 \dots n \quad \text{fint}_i := f\left(x_i + \frac{h}{2}\right) \quad I := h \cdot \sum_{j=1}^n \text{fint}_j \quad I = 0.5016835.$$

Подпрограмма вычислений по формуле парабол (4.5.6) может быть такой:

$$\text{parab}(f, a, b, n) := \begin{cases} h \leftarrow \frac{(b - a)}{n} \\ \text{for } i \in 1..n \\ \quad x_i \leftarrow a + h(i - 1) \\ \quad f2_i \leftarrow f(x_i) \\ \quad f1_i \leftarrow f\left(x_i + \frac{h}{2}\right) \\ f2_{n+1} \leftarrow f(b) \\ I \leftarrow \frac{h}{6} \left(f2_1 + f2_{n+1} + 4 \sum_{i=1}^n f1_i + 2 \sum_{i=2}^n f2_i \right) \\ I \end{cases}$$

$$I1 := \text{parab}(f, a, b, n) \quad I1 = 0.5016991 \quad I2 := \text{parab}(f1, a, b, n) \quad I2 = 0.9524545$$

Ее параметры: f - интегрируемая функция, которая должна быть ранее определена в программе, a, b - пределы интегрирования, n - число точек сетки (четное число для формулы парабол).

Задание № 2. Найти численное значение интеграла от функций, указанных в задании № 1, на заданном промежутке. При вычислении по подпрограмме parab выбрать n , соответствующее меньшему значению шага.

5. РЕШЕНИЕ СИСТЕМ ЛИНЕЙНЫХ АЛГЕБРАИЧЕСКИХ УРАВНЕНИЙ И ПРОБЛЕМЫ СОБСТВЕННЫХ ЗНАЧЕНИЙ

5.1. Нормы векторов и матриц и их свойства

В вычислительной линейной алгебре выделяют четыре основные задачи:

- 1) решение систем линейных алгебраических уравнений;
- 2) вычисление определителей;
- 3) нахождение обратных матриц;
- 4) определение собственных значений и собственных векторов.

Рассмотрим подробнее первую задачу. Пусть дана система линейных уравнений

$$\begin{cases} a_{11}x_1 + a_{12}x_2 + \dots + a_{1n}x_n = b_1, \\ a_{21}x_1 + a_{22}x_2 + \dots + a_{2n}x_n = b_2, \\ \dots \\ a_{n1}x_1 + a_{n2}x_2 + \dots + a_{nn}x_n = b_n. \end{cases} \quad (5.1.1)$$

В матричной форме записи эта система принимает вид

$$A\bar{x} = \bar{b}, \quad (5.1.2)$$

где

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

Если матрица A не является вырожденной, то решение системы существует, единственно и устойчиво ко входным данным.

Пусть $\bar{x}^* = (x_1^*, x_2^*, \dots, x_n^*)^T$ - приближенное решение системы (5.1.1). Тогда $\bar{e} = \bar{x}_{\text{точн.}} - \bar{x}^*$ - погрешность решения. Кроме ее качества полученного решения часто можно оценить по невязке $\bar{r} = \bar{b} - A\bar{x}^*$,

которая показывает, насколько правая часть системы отличается от левой, если подставить в нее приближенное решение $\bar{r} = A\bar{x}_{\text{точн.}} - A\bar{x}^* = A(\bar{x}_{\text{точн.}} - \bar{x}^*)$. Тогда

$$\bar{r} = A\bar{e}, \quad \bar{e} = A^{-1}\bar{r}. \quad (5.1.4)$$

Для того чтобы анализировать методы решения систем, необходимо уметь количественно оценивать «величины» векторов \bar{x}^* и $\bar{x}_{\text{точн.}} - \bar{x}^*$, а также векторов \bar{b} и $\bar{b} - \bar{b}^*$. Для этой цели используют понятие нормы вектора.

Говорят, что в R^n задана норма, если $\forall \bar{x} \in R^n$ сопоставлено число $\|\bar{x}\| \in R$, называемое нормой вектора \bar{x} и обладающее следующими свойствами:

- 1) $\|\bar{x}\| > 0$, если $\|\bar{x}\| = 0 \Leftrightarrow \bar{x} = 0$,
- 2) $\|\alpha\bar{x}\| = |\alpha| \cdot \|\bar{x}\|$, $\forall \bar{x} \in R^n$ и $\forall \alpha \in R$,
- 3) $\|\bar{x} + \bar{y}\| \leq \|\bar{x}\| + \|\bar{y}\|$ - неравенство треугольника.

Существует множество различных способов введения норм. В вычислительных методах наиболее употребительными являются три нормы:

$$\|\bar{x}\|_1 = \sum_{i=1}^n |x_i|, \quad (5.1.5)$$

$$\|\bar{x}\|_2 = \sqrt{\sum_{i=1}^n |x_i|^2}, \quad (5.1.6)$$

$$\|\bar{x}\|_{\infty} = \max_{1 \leq i \leq n} |x_i|. \quad (5.1.7)$$

При этом $\|\bar{x}\|_{\infty} \leq \|\bar{x}\|_2 \leq \|\bar{x}\|_1 \leq n \cdot \|\bar{x}\|_{\infty}$. Норма (5.1.6) является естественным обобщением на случай n -мерного пространства понятия длины вектора в двух и трехмерных геометрических пространствах. Поэтому ее называют евклидовой нормой.

Абсолютной и относительной погрешностью вектора \bar{x}^* называются выражения

$$\Delta(\bar{x}^*) = \|\bar{x} - \bar{x}^*\|, \quad \delta(\bar{x}^*) = \frac{\|\bar{x} - \bar{x}^*\|}{\|\bar{x}\|}. \quad (5.1.8)$$

Выбор той или иной конкретной нормы в практических задачах диктуется тем, какие требования предъявляются к точности решения. Выбор нормы (5.1.5) отвечает случаю, когда малой должна быть суммарная абсолютная ошибка в компонентах решения; выбор (5.1.6) соответствует критерию малости среднеквадратической ошибки, а выбор (5.1.7) означает, что малой должна быть максимальная из абсолютных ошибок в компонентах решения.

Величина $\|A\| = \max_{x \neq 0} \frac{\|Ax\|}{\|\bar{x}\|}$ называется нормой матрицы A , подчиненной норме вектора \bar{x} , введенной в R^n . Норма матрицы обладает следующими свойствами:

1) $\|A\| \geq 0$, $\|A\| = 0 \Leftrightarrow A = 0$,

2) $\|\alpha A\| = |\alpha| \cdot \|A\|$,

3) $\|A + B\| \leq \|A\| + \|B\|$,

4) $\|A \cdot B\| \leq \|A\| \cdot \|B\|$,

5) $\|A\bar{x}\| \leq \|A\| \cdot \|\bar{x}\|$.

Как следует из определения, каждой из векторных норм $\|\bar{x}\|$ соответствует своя подчиненная норма матрицы A . Нормам $\|\bar{x}\|_1, \|\bar{x}\|_2, \|\bar{x}\|_{\infty}$ подчинены нормы $\|A\|_1, \|A\|_2$ и $\|A\|_{\infty}$, вычисляемые по формулам

$$\|A\|_1 = \max_{1 \leq j \leq n} \sum_{i=1}^n |a_{ij}|, \quad (5.1.9)$$

$$\|A\|_2 = \max_{1 \leq j \leq n} \sqrt{\lambda_j(A^T A)}, \quad (5.1.10)$$

где $\lambda_j(A^T A)$ - собственные числа матрицы $A^T A$,

$$\|A\|_{\infty} = \max_{1 \leq i \leq n} \sum_{j=1}^n |a_{ij}|. \quad (5.1.11)$$

Так как вычисление нормы (5.1.10) затруднено, то для приближенной оценки величины $\|A\|_2$ можно использовать неравенство

$$\|A\|_2 \leq \|A\|_E = \sqrt{\sum_{i,j=1}^n |a_{ij}|^2}, \quad (5.1.12)$$

где $\|A\|_E$ - евклидова норма матрицы A .

Норма матрицы имеет простую геометрическую интерпретацию. Операцию умножения матрицы A на вектор \bar{x} можно рассматривать как преобразование \bar{x} в новый вектор

$\bar{y} = A\bar{x}$. Так как $\|\bar{x}\|$ - длина \bar{x} , то $\frac{\|A\bar{x}\|}{\|\bar{x}\|}$ есть коэффициент растяжения вектора \bar{x} под действием матрицы A .

5.2. Обусловленность задачи решения системы линейных алгебраических уравнений

Рассмотрим систему $A\bar{x} = \bar{b}$. Пусть A задана совершенно точно, а вектор-столбец \bar{b} - приближенно.

Теорема 5.1. Пусть \tilde{x} - точное решение системы $A\bar{x} = \bar{b}^*$, в которой правая часть \bar{b}^* является приближением к \bar{b} . Тогда верны следующие оценки абсолютной и относительной погрешностей:

$$\begin{aligned}\Delta(\tilde{x}) &\leq \vartheta_\Delta \Delta(\bar{b}^*) \\ \delta(\tilde{x}) &\leq \vartheta_\delta \delta(\bar{b}^*)\end{aligned}\quad (5.2.1)$$

где $\vartheta_\Delta = \|A^{-1}\|$, $\vartheta_\delta = \|A^{-1}\| \cdot \frac{\|\bar{b}\|}{\|\bar{x}\|}$.

Действительно, $\Delta(\tilde{x}) = \|\bar{x} - \tilde{x}\|$, $\delta(\tilde{x}) = \frac{\|\bar{x} - \tilde{x}\|}{\|\bar{x}\|}$, $\bar{r} = A(\bar{x} - \tilde{x}) = \bar{b} - \bar{b}^*$. Тогда

$$\begin{aligned}\Delta(\tilde{x}) &\leq \|A^{-1}\| \cdot \|\bar{r}\|, \quad \Delta(\tilde{x}) \leq \|A^{-1}\| \cdot \|\bar{b} - \bar{b}^*\| = \|A^{-1}\| \Delta(\bar{b}^*) \\ \delta(\tilde{x}) &= \frac{\Delta(\tilde{x})}{\|\bar{x}\|} \leq \frac{\|A^{-1}\| \Delta(\bar{b}^*)}{\|\bar{x}\|} \cdot \frac{\|\bar{b}\|}{\|\bar{b}\|} = \\ &= \frac{\|A^{-1}\| \cdot \|\bar{b}\|}{\|\bar{x}\|} \Delta(\bar{b}^*) = \vartheta_\delta \delta(\bar{b}^*)\end{aligned}$$

Величина ϑ_Δ для системы $A\bar{x} = \bar{b}$ играет роль абсолютного числа обусловленности; величина $\vartheta_\delta = \frac{\|A^{-1}\| \cdot \|\bar{b}\|}{\|\bar{x}\|}$ называется естественным числом обусловленности и характеризует коэффициент возможного возрастания относительной погрешности решения, вызванной погрешностью правой части.

Вычислим максимальное значение ϑ_δ . Тогда $\max_{x \neq 0} \vartheta_\delta(x) = \max_{x \neq 0} \frac{\|A^{-1}\| \cdot \|Ax\|}{\|x\|} = \|A^{-1}\| \cdot \|A\|$.

Полученная величина называется стандартным числом обусловленности и обозначается

$$\vartheta(A) = \text{cond}(A) = \|A^{-1}\| \cdot \|A\|. \quad (5.2.2)$$

Поскольку $\text{cond}(A) = \max \vartheta_\delta$, то $\delta(\tilde{x}) \leq \text{cond}(A) \cdot \delta(\bar{b}^*)$.

Теорема 5.2. Пусть \tilde{x} - точное решение системы $A^* \bar{x} = \bar{b}$ с приближенно заданной матрицей A^* . Тогда верна следующая оценка относительной точности:

$$\begin{aligned}\delta^*(\tilde{x}) &\leq \text{cond}(A) \cdot \delta(A^*), \quad \text{где} \\ \delta^*(\tilde{x}) &= \frac{\|\bar{x} - \tilde{x}\|}{\|\tilde{x}\|}, \quad \delta(A^*) = \frac{\|A - A^*\|}{\|A\|}.\end{aligned}\quad (5.2.3)$$

В случае, когда с погрешностью заданы как правая часть системы, так и матрица A , можно доказать справедливость неравенства

$$\delta(\tilde{x}) \leq \text{cond}(A) \cdot (\delta(\bar{b}^*) + \delta(A^*)) \quad (5.2.4)$$

В заключение заметим, что для вычисления $\text{cond}(A)$ необходимо иметь A^{-1} . Операция вычисления обратной матрицы очень трудоемка, требует примерно $2n^3$ операций, поэтому на практике избегают такого способа вычисления $\text{cond}(A)$.

5.3. Метод Гаусса (схема единственного деления)

Вычисления с помощью метода Гаусса состоят из двух основных этапов, называемых прямым и обратным ходом. Прямой ход заключается в последовательном исключении неизвестных из системы $A\bar{x} = \bar{b}$ для преобразования ее к эквивалентной системе с верхней треугольной матрицей. Вычисления значений неизвестных происходят на этапе обратного хода. Прямой ход состоит из $n - 1$ шагов.

1-й шаг. Целью этого шага является исключение неизвестного x_1 из уравнений с номерами $i = 2, 3, \dots, n$. Пусть $a_{11} \neq 0$. Тогда этот элемент называется главным (ведущим) элементом первого шага. Найдем $\mu_{1i} = \frac{a_{i1}}{a_{11}}$, $i = 2, 3, \dots, n$. Вычтем последовательно из второго,

третьего,..., n -го уравнения системы (5.1.1) первое уравнение, умноженное соответственно на $\mu_{21}, \mu_{31}, \dots, \mu_{n1}$. Это позволит обратить в нуль коэффициенты при x_1 во всех уравнениях, кроме первого. В результате будет получена эквивалентная система (5.3.1), в которой $a_{ij}^{(1)} = a_{ij} - \mu_{1i}a_{1j}$, $b_i^{(1)} = b_i - \mu_{1i}b_1$:

$$\left\{ \begin{array}{l} a_{11}x_1 + a_{12}x_2 + a_{13}x_3 + \dots + a_{1n}x_n = b_1, \\ a_{22}^{(1)}x_2 + a_{23}^{(1)}x_3 + \dots + a_{2n}^{(1)}x_n = b_2^{(1)}, \\ a_{32}^{(1)}x_2 + a_{33}^{(1)}x_3 + \dots + a_{3n}^{(1)}x_n = b_3^{(1)}, \\ \dots \\ a_{n2}^{(1)}x_2 + a_{n3}^{(1)}x_3 + \dots + a_{nn}^{(1)}x_n = b_n^{(1)}. \end{array} \right. \quad (5.3.1)$$

2-й шаг. Целью этого шага является исключение неизвестного x_2 из уравнений с номерами $i = 3, 4, \dots, n$. Пусть $a_{22}^{(1)} \neq 0$ - ведущий элемент второго шага; положим опять $\mu_{i2} = \frac{a_{i2}^{(1)}}{a_{22}^{(1)}}$ и вычтем из третьего, четвертого,..., n -го уравнений второе уравнение, умноженное на $\mu_{32}, \mu_{42}, \dots, \mu_{n2}$ соответственно. Получим

$$\left\{ \begin{array}{l} a_{11}x_1 + a_{12}x_2 + a_{13}x_3 + \dots + a_{1n}x_n = b_1, \\ a_{22}^{(1)}x_2 + a_{23}^{(1)}x_3 + \dots + a_{2n}^{(1)}x_n = b_2^{(1)}, \\ a_{33}^{(2)}x_3 + \dots + a_{3n}^{(2)}x_n = b_3^{(2)}, \\ \dots \\ a_{n3}^{(2)}x_3 + \dots + a_{nn}^{(2)}x_n = b_n^{(2)}, \end{array} \right. \quad (5.3.2)$$

где $a_{ij}^{(2)} = a_{ij}^{(1)} - \mu_{i2}a_{2j}^{(1)}$, $b_i^{(2)} = b_i^{(1)} - \mu_{i2}b_2^{(1)}$. В результате после $n - 1$ -го шага исключения получим следующую треугольную систему уравнений:

$$\left\{ \begin{array}{l} a_{11}x_1 + a_{12}x_2 + a_{13}x_3 + \dots + a_{1n}x_n = b_1, \\ a_{22}^{(1)}x_2 + a_{23}^{(1)}x_3 + \dots + a_{2n}^{(1)}x_n = b_2^{(1)}, \\ a_{33}^{(2)}x_3 + \dots + a_{3n}^{(2)}x_n = b_3^{(2)}, \\ \dots \\ a_{nn}^{(n-1)}x_n = b_n^{(n-1)}. \end{array} \right. \quad (5.3.3)$$

На этом вычисления прямого хода заканчиваются.

Обратный ход посвящен нахождению неизвестных x_1, x_2, \dots, x_n . Из последнего уравнения системы (5.3.3) находим сразу $x_n = \frac{b_n^{(n-1)}}{a_{nn}^{(n-1)}}$. Подставляя найденное значение x_n в предпоследнее уравнение, получим x_{n-1} . Осуществляя обратную подстановку, далее последовательно находим $x_{n-2}, x_{n-3}, \dots, x_1$.

Общее число арифметических операций прямого хода в методе Гаусса примерно $\frac{2}{3}n^3$, обратного - всего около n^2 , что при большом n пренебрежимо мало по сравнению с числом операций прямого хода.

Заметим, что вычисление множителей, а также обратная подстановка требуют деления на главные элементы $a_{ii}^{(i-1)}$. Поэтому если один из главных элементов оказывается близким к нулю, то схема единственного деления в уже описанном виде не может быть реализована. В этом случае прибегают к выбору главного элемента по столбцу (схема частичного выбора) или в выбору главного элемента по всей матрице (схема полного выбора).

В схеме частичного выбора главного элемента на каждом k -м шаге исключения выбирается максимальный по модулю коэффициент $a_{i_k,k}$ при неизвестной x_k в уравнениях с номерами $i = k, k+1, \dots, n$. Этим гарантируется, что $|\mu_{ik}| \leq 1$ для всех переменных $k = 1, 2, \dots, n-1$ и уравнений $i = k+1, \dots, n$.

В схеме главного выбора допускается нарушение естественного порядка исключения неизвестных. Здесь на первом шаге среди всех элементов a_{ij} определяется максимальный по модулю элемент a_{i_1,j_1} . Первое уравнение системы и уравнение с номером i_1 меняются местами. Затем производится исключение неизвестного x_{j_1} из всех уравнений, кроме первого. На всех других шагах последовательность действий аналогична.

В качестве **примера** рассмотрим следующую систему:

$$\begin{cases} 1.2x_1 + 1.8x_2 - 2.2x_3 - 4.1x_4 = 1.3, \\ 10.0x_1 - 5.1x_2 + 1.2x_3 + 5.5x_4 = 1.2, \\ 2.2x_1 - 30.1x_2 + 3.1x_3 + 5.8x_4 = 10.0, \\ 10.0x_1 - 2.4x_2 - 30.5x_3 - 2.2x_4 = 34.1. \end{cases}$$

Решим ее по схеме полного выбора с $\varepsilon = 10^{-3}$. Максимальный по модулю элемент $a_{34} = -30.5$ содержится в четвертом уравнении, поэтому переставим его на первое место. Исключаем неизвестное x_3 из второго, третьего и четвертого уравнений, при этом

$$\mu_{23} = \frac{-2.2}{-30.5} = 0.0721, \quad \mu_{33} = -0.0393, \quad \mu_{43} = -0.1016 :$$

$$\begin{cases} 10.0000x_1 - 2.4000x_2 - 30.5000x_3 - 2.2000x_4 = 34.1000, \\ 0.4790x_1 + 1.9730x_2 - 3.9414x_4 = -1.1586, \\ 10.3930x_1 - 5.1943x_2 + 5.4135x_4 = 2.5401, \\ 3.2160x_1 - 30.3438x_2 + 5.5765x_4 = 13.4646. \end{cases}$$

Среди коэффициентов при неизвестных во втором, третьем и четвертом уравнениях максимальным по модулю является коэффициент $a_{24}^{(1)} = -30.3438$. Переставим четвертое уравнение на место второго и исключим x_2 :

$$\begin{cases} 10.0000x_1 - 2.4000x_2 - 30.5000x_3 - 2.2000x_4 = 34.1000, \\ 3.2160x_1 - 30.3438x_2 + 5.5765x_4 = 13.4646, \\ 0.6880x_1 - 3.5789x_4 = -0.2834, \\ 9.8424x_1 + 4.4588x_4 = 0.2350. \end{cases}$$

В третьем и четвертом уравнениях наибольший коэффициент $a_{14}^{(2)} = 9.8424$. Поэтому переставим четвертое уравнение на место третьего и исключим x_1 . Тогда получим следующую систему:

$$\begin{cases} 10.0000x_1 - 2.4000x_2 - 30.5000x_3 - 2.2000x_4 = 34.1000, \\ 3.2160x_1 - 30.3438x_2 + 5.5765x_4 = 13.4646, \\ 9.8424x_1 + 4.4588x_4 = 0.2350, \\ 9.8424x_1 + 3.8906x_4 = 0.2998. \end{cases}$$

Прямой ход закончен. Вычислим неизвестные при помощи обратного хода. Последовательно получим

$$\begin{aligned} x_4 &= 0.077, \quad x_1 = \frac{(0.2350 - 0.3438)}{9.8424} = -0.011, \\ x_2 &= \frac{(13.4646 - 0.4299 + 0.0354)}{-30.3438} = -0.431, \\ x_3 &= \frac{(34.1000 + 0.1850 + 0.1110 - 1.0337)}{-30.5000} = -1.094. \end{aligned}$$

Итак, с точностью $\epsilon = 10^{-3}$

$$\begin{cases} x_1 = -0.011, \\ x_2 = -0.431, \\ x_3 = -1.094, \\ x_4 = 0.077. \end{cases}$$

5.4. Метод прогонки

Метод прогонки, так же как и метод Гаусса, разделяется на два этапа: прямой и обратный ход. В результате выполнения прямого хода вычисляются вспомогательные переменные - так называемые прогоночные коэффициенты. Обратный ход дает значения неизвестных.

Метод прогонки специально создан для решения линейных алгебраических уравнений с трехдиагональной матрицей, то есть для систем вида

$$\begin{cases} b_1x_1 + c_1x_2 = d_1, \\ a_2x_1 + b_2x_2 + c_2x_3 = d_2, \\ \dots \\ a_{n-1}x_{n-2} + b_{n-1}x_{n-1} + c_{n-1}x_n = d_{n-1}, \\ a_nx_{n-1} + b_nx_n = d_n. \end{cases} \quad (5.4.1)$$

Системы такого вида часто возникают при решении различных задач математической физики и интерполяции сплайнами.

Выведем формулы метода. Из первого уравнения $b_1x_1 + c_1x_2 = d_1$ найдем x_1 :

$$x_1 = \frac{d_1 - c_1x_2}{b_1} = \frac{d_1}{b_1} - \frac{c_1}{b_1}x_2 = \alpha_1x_2 + \beta_1. \quad (5.4.2)$$

Подставим полученное выражение для x_1 во второе уравнение системы (5.4.1):

$$a_2(\alpha_1x_2 + \beta_1) + b_2x_2 + c_2x_3 = d_2, \quad a_2\alpha_1x_2 + b_2x_2 = d_2 - a_2\beta_1 - c_2x_3,$$

$$x_2 = -\frac{c_2}{b_2 + a_2 \alpha_1} x_3 + \frac{d_2 - a_2 \beta_1}{b_2 + a_2 \alpha_1} = \alpha_2 x_3 + \beta_2. \quad (5.4.3)$$

Последнее уравнение для x_2 подставим в третье уравнение системы (5.4.1) и так далее.

На i -м шаге процесса ($1 < i < n$) i -е уравнение системы преобразуется к такому же виду:

$$\begin{aligned} x_i &= \alpha_i x_{i+1} + \beta_i, \text{ где} \\ \alpha_i &= -\frac{c_i}{b_i + a_i \alpha_{i-1}}, \quad \beta_i = \frac{d_i - a_i \beta_{i-1}}{b_i + a_i \alpha_{i-1}}. \end{aligned} \quad (5.4.4)$$

На последнем n -м шаге подстановка в последнее уравнение системы $x_{n-1} = \alpha_{n-1} x_n + \beta_{n-1}$ даст $a_n(\alpha_{n-1} x_n + \beta_{n-1}) + b_n x_n = d_n$. Отсюда

$$x_n = \frac{d_n - a_n \beta_{n-1}}{b_n + a_n \alpha_{n-1}}. \quad (5.4.5)$$

Значения остальных неизвестных вычисляются в процессе обратного хода по формулам (5.4.4). Итак, прямой ход метода прогонки состоит в вычислении прогоночных коэффициентов

$$\left\{ \begin{array}{l} i = 1 : \alpha_1 = -\frac{c_1}{\gamma_1}, \quad \beta_1 = \frac{d_1}{\gamma_1}, \quad \gamma_1 = b_1, \\ i = 2, 3, \dots, n-1 : \alpha_i = -\frac{c_i}{\gamma_i}, \quad \beta_i = \frac{d_i - a_i \beta_{i-1}}{\gamma_i}, \quad \gamma_i = b_i + a_i \alpha_{i-1}, \\ i = n : \beta_n = \frac{d_n - a_n \beta_{n-1}}{\gamma_n}, \quad \gamma_n = b_n + a_n \alpha_{n-1}. \end{array} \right. \quad (5.4.6)$$

Обратный ход метода прогонки дает значения неизвестных по формулам

$$x_n = \beta_n, \quad x_i = \alpha_i x_{i+1} + \beta_i, \quad i = n-1, n-2, \dots, 2, 1. \quad (5.4.7)$$

Несложный подсчет показывает, что для реализации вычислений по описанному алгоритму требуется примерно $8n$ арифметических операций, что гораздо меньше числа операций в методе Гаусса $\approx \frac{2}{3}n^3$. Кроме того структура матрицы коэффициентов системы (5.4.1) позволяет использовать для ее хранения лишь $3n - 2$ машинных слова.

Теоретические исследования показали, что для существования решения системы (5.4.1) и его единственности необходимо выполнение следующих условий.

Теорема 5.3. Пусть коэффициенты системы (5.4.1) удовлетворяют условиям **диагонального преобладания**: $|b_k| \geq |a_k| + |c_k|$, $|b_k| > |a_k|$, $1 \leq k \leq n$, тогда $\gamma_i \neq 0$ и $|\alpha_i| \leq 1$ для всех $i = \overline{1, n}$.

Условие $\gamma_i \neq 0$ и $|\alpha_i| \leq 1$ для всех $i = \overline{1, n}$ означает, что метод прогонки устойчив по входным данным.

В качестве **примера** решим следующую систему, уже встречавшуюся нам при рассмотрении кубических сплайнов:

$$\left\{ \begin{array}{l} x_1 - x_2 = 0.8, \\ x_1 + 4x_2 + x_3 = 3.6, \\ x_2 + 4x_3 + x_4 = -1.2, \\ x_3 + 4x_4 + x_5 = -3.6, \\ x_4 - x_5 = -0.8. \end{array} \right.$$

Прямой ход. $i = 1$, $\gamma_1 = b_1 = 1$, $\alpha_1 = -\frac{c_1}{\gamma_1} = 1$, $\beta_1 = \frac{d_1}{\gamma_1} = 0.8$,

$i = 2$, $\gamma_2 = b_2 + a_2 \alpha_1 = 4 + 1 = 5$, $\alpha_2 = -\frac{c_2}{\gamma_2} = -\frac{1}{5} = -0.2$, $\beta_2 = \frac{d_2 - a_2 \beta_1}{\gamma_2} = \frac{3.6 - 0.8}{5} = 0.56$.

$$i = 3, \gamma_3 = b_3 + a_3\alpha_2 = 4 - 0.2 = 3.8, \alpha_3 = -\frac{c_3}{\gamma_3} = -\frac{1}{3.8} = -0.263158,$$

$$\beta_3 = \frac{d_3 - a_3\beta_2}{\gamma_3} = \frac{-1.2 - 0.56}{3.8} = -0.463158.$$

$$i = 4, \gamma_4 = b_4 + a_4\alpha_3 = 4 - 0.263158 = 3.736842, \alpha_4 = -\frac{c_4}{\gamma_4} = -0.267606,$$

$$\beta_4 = \frac{d_4 - a_4\beta_3}{\gamma_4} = \frac{-3.6 + 0.463158}{3.736842} = -0.839437.$$

$$i = 5, \gamma_5 = b_5 + a_5\alpha_4 = -1 - 0.267606 = -1.267606, \beta_5 = \frac{d_5 - a_5\beta_4}{\gamma_5} = \frac{-0.8 + 0.839437}{-1.267606} = -0.031111.$$

Обратный ход. $x_5 = \beta_5 = -0.031111$,

$$x_4 = \alpha_4 x_5 + \beta_4 = (-0.267606) \cdot (-0.031111) + (-0.839457) = -0.831132,$$

$$x_3 = \alpha_3 x_4 + \beta_3 = (-0.263158) \cdot (-0.831132) + (-0.463158) = -0.244439,$$

$$x_2 = \alpha_2 x_3 + \beta_2 = (-0.200000) \cdot (-0.244439) + 0.560000 = 0.608888,$$

$$x_1 = \alpha_1 x_2 + \beta_1 = 1.000000 \cdot 0.608888 + 0.800000 = 1.408888.$$

5.5. Метод простых итераций

Итерационные методы применяют главным образом для решения задач большой размерности, когда использование прямых методов невозможно из-за ограничений технического характера. Большие системы уравнений, возникающие в приложениях, как правило, являются разреженными. При использовании метода Гаусса, например, большое число нулевых элементов превращаются в ненулевые и матрица теряет свойство разреженности. Использование итерационных методов не меняет матрицу коэффициентов, она остается разреженной.

Однако применение итерационных методов для качественного решения требует серьезного использования структуры системы уравнений, специальных знаний и опыта.

Пусть дана система $\bar{A}\bar{x} = \bar{b}$, где A - квадратная невырожденная матрица. Преобразуем ее к виду $\bar{x} = \bar{B}\bar{x} + \bar{c}$, (5.5.1)

где B - квадратная матрица такой же размерности что и A , \bar{c} - вектор -столбец. В развернутой форме записи системы (5.5.1) имеет вид

$$\begin{cases} x_1 = b_{11}x_1 + b_{12}x_2 + \dots + b_{1n}x_n + c_1, \\ x_2 = b_{21}x_1 + b_{22}x_2 + \dots + b_{2n}x_n + c_2, \\ \dots \\ x_n = b_{n1}x_1 + b_{n2}x_2 + \dots + b_{nn}x_n + c_n. \end{cases} \quad (5.5.2)$$

Операция приведения системы $\bar{A}\bar{x} = \bar{b}$ к виду (5.5.2) не является очевидной и простой и требует специальных знаний, а также существенного использования специфики системы. Самый простой способ приведения системы $\bar{A}\bar{x} = \bar{b}$ к виду (5.5.2) состоит в последовательном исключении из первого уравнения системы $\bar{A}\bar{x} = \bar{b}$ переменной x_1 , из второго уравнения - переменной x_2 и так далее. Метод итерации в такой реализации называется методом Якоби*. Система уравнений метода Якоби имеет вид

* Карл Густав Якоб Якоби (1804-1851) - немецкий математик.

$$\begin{cases} x_1 = b_{12}x_2 + b_{13}x_3 + \dots + b_{1n-1}x_{n-1} + b_{1n}x_n + c_1, \\ x_2 = b_{21}x_1 + b_{23}x_3 + \dots + b_{2n-1}x_{n-1} + b_{2n}x_n + c_2, \\ \dots \\ x_n = b_{n1}x_1 + b_{n2}x_2 + b_{n3}x_3 + \dots + b_{nn-1}x_{n-1} + c_n. \end{cases} \quad (5.5.3)$$

На главной диагонали матрицы B системы (5.5.3) стоят нули, а остальные элементы, очевидно, выражаются по формулам $b_{ij} = -\frac{a_{ij}}{a_{ii}}$, $c_i = \frac{b_i}{a_{ii}}$, $i, j = \overline{1, n}$, $i \neq j$.

Практически метод работает следующим способом. Выбирается начальное приближение $\bar{x}^{(0)} = (x_1^{(0)}, x_2^{(0)}, \dots, x_n^{(0)})^T$ и подставляется в правую часть системы (5.5.1). Решая систему, находят первое приближение $\bar{x}^{(1)} = B\bar{x}^{(0)} + \bar{c}$. Это приближение опять подставляют в правую часть (5.5.1). Таким образом, получается $\bar{x}^{(2)} = B\bar{x}^{(1)} + \bar{c}$. Продолжая этот процесс далее, получим последовательность $\bar{x}^{(0)}, \bar{x}^{(1)}, \bar{x}^{(2)}, \dots, \bar{x}^{(n)}$... приближений, вычисляемых по формуле $\bar{x}^{(k+1)} = B\bar{x}^{(k)} + \bar{c}$, $k = 0, 1, 2, \dots$ (5.5.4)

В развернутой форме записи система (5.5.4) выглядит таким образом:

$$\begin{cases} x_1^{(k+1)} = b_{11}x_1^{(k)} + b_{12}x_2^{(k)} + \dots + b_{1n}x_n^{(k)} + c_1, \\ x_2^{(k+1)} = b_{21}x_1^{(k)} + b_{22}x_2^{(k)} + \dots + b_{2n}x_n^{(k)} + c_2, \\ \dots \\ x_n^{(k+1)} = b_{n1}x_1^{(k)} + b_{n2}x_2^{(k)} + \dots + b_{nn}x_n^{(k)} + c_n. \end{cases} \quad (5.5.5)$$

5.6. Сходимость метода простых итераций

Теорема 5.4. Пусть $\|B\| < 1$. Тогда решение $\bar{x}_{\text{точн.}}$ системы $\bar{x} = B\bar{x} + \bar{c}$ существует и единственno. При любом начальном приближении $\bar{x}^{(0)}$ метод простых итераций сходится и справедлива оценка погрешности

$$\left\| \bar{x}^{(n)} - \bar{x}_{\text{точн.}} \right\| \leq \|B\|^n \left\| \bar{x}^{(0)} - \bar{x}_{\text{точн.}} \right\|. \quad (5.6.1)$$

Доказательство

Пусть в (5.5.5) $k+1 = n$, тогда $\bar{x}^{(n)} = B\bar{x}^{(n-1)} + \bar{c}$. Если $\bar{x}_{\text{точн.}}$ - точное решение системы, то оно удовлетворяет уравнению (5.5.1), то есть $\bar{x}_{\text{точн.}} = B\bar{x}_{\text{точн.}} + \bar{c}$. Вычтем два последних уравнения друг из друга. Получим $\bar{x}^{(n)} - \bar{x}_{\text{точн.}} = B(\bar{x}^{(n-1)} - \bar{x}_{\text{точн.}})$. Найдем норму этого выражения: $\left\| \bar{x}^{(n)} - \bar{x}_{\text{точн.}} \right\| = \left\| B(\bar{x}^{(n-1)} - \bar{x}_{\text{точн.}}) \right\| \leq \|B\| \cdot \left\| \bar{x}^{(n-1)} - \bar{x}_{\text{точн.}} \right\| \leq \|B\| \cdot \|B\| \cdot \left\| \bar{x}^{(n-2)} - \bar{x}_{\text{точн.}} \right\| \leq \dots \leq \|B\|^n \cdot \left\| \bar{x}^{(0)} - \bar{x}_{\text{точн.}} \right\|$, так как неравенство верно для всех индексов от 0 до n .

Итак, метод простых итераций сходится со скоростью геометрической прогрессии, знаменатель которой $q = \|B\|$. Скорость сходимости тем выше, чем меньше величина $\|B\|$. Хотя метод сходится при любом начальном приближении $\bar{x}^{(0)}$, ясно, что начальное приближение нужно выбирать ближе к точному решению. Приведенная в теореме 5.4 оценка точности решения является априорной. Ее практическое использование затруднительно, так как $\left\| \bar{x}^{(0)} - \bar{x}_{\text{точн.}} \right\|$ неизвестно, а его грубое оценивание заведомо приведет к завышению числа итераций n .

Теорема 5.5. (Апостериорная оценка погрешности решения).

Если $\|B\| < 1$, то справедлива следующая оценка:

$$\left\| \bar{x}^{(n)} - \bar{x}_{\text{точн.}} \right\| \leq \frac{\|B\|}{1 - \|B\|} \left\| \bar{x}^{(n)} - \bar{x}^{(n-1)} \right\|. \quad (5.6.2)$$

Доказательство

В предыдущей теореме имели равенство $\bar{x}^{(n)} - \bar{x}_{\text{точн.}} = B(\bar{x}^{(n-1)} - \bar{x}_{\text{точн.}})$. Преобразуем его алгебраически: $\bar{x}^{(n)} - \bar{x}_{\text{точн.}} = B(\bar{x}^{(n-1)} - \bar{x}^{(n)} + \bar{x}^{(n)} - \bar{x}_{\text{точн.}}) = B(\bar{x}^{(n-1)} - \bar{x}^{(n)}) + B(\bar{x}^{(n)} - \bar{x}_{\text{точн.}})$. Тогда $\left\| \bar{x}^{(n)} - \bar{x}_{\text{точн.}} \right\| \leq \|B\| \cdot \left\| \bar{x}^{(n-1)} - \bar{x}^{(n)} \right\| + \|B\| \cdot \left\| \bar{x}^{(n)} - \bar{x}_{\text{точн.}} \right\|$. Отсюда легко получаем

$$\left\| \bar{x}^{(n)} - \bar{x}_{\text{точн.}} \right\| \leq \frac{\|B\|}{1 - \|B\|} \left\| \bar{x}^{(n)} - \bar{x}^{(n-1)} \right\|.$$

Если требуется найти решение с точностью ε , то следует проводить итерации до выполнения неравенства $\frac{\|B\|}{1 - \|B\|} \left\| \bar{x}^{(n)} - \bar{x}^{(n-1)} \right\| < \varepsilon$. Таким образом, в качестве критерия окончания итерационного процесса может быть использовано неравенство

$$\left\| \bar{x}^{(n)} - \bar{x}^{(n-1)} \right\| < \varepsilon_1, \text{ где } \varepsilon_1 = \frac{1 - \|B\|}{\|B\|} \varepsilon.$$

5.7. Метод Зейделя*

Пусть система $A\bar{x} = \bar{b}$ методом Якоби приведена к виду (5.5.3):

$$\begin{cases} x_1 = b_{12}x_2 + b_{13}x_3 + \dots + b_{1n-1}x_{n-1} + b_{1n}x_n + c_1, \\ x_2 = b_{21}x_1 + b_{23}x_3 + \dots + b_{2n-1}x_{n-1} + b_{2n}x_n + c_2, \\ \dots \\ x_n = b_{n1}x_1 + b_{n2}x_2 + b_{3n}x_3 + \dots + b_{nn-1}x_{n-1} + c_n. \end{cases}$$

Метод Зейделя является лишь модификацией метода Якоби. Его основная идея состоит в том, что при вычислении очередного $(k+1)$ -го приближения к неизвестному x_i при $i > 1$ используют уже найденные $(k+1)$ -е приближения к неизвестным x_1, x_2, \dots, x_{i-1} , а не k -е приближение, как в методе Якоби. Система (5.5.3) на $(k+1)$ -й итерации будет выглядеть так:

$$\begin{cases} x_1^{(k+1)} = b_{12}x_2^{(k)} + b_{13}x_3^{(k)} + \dots + b_{1n}x_n^{(k)} + c_1, \\ x_2^{(k+1)} = b_{21}x_1^{(k+1)} + b_{23}x_3^{(k)} + \dots + b_{2n}x_n^{(k)} + c_2, \\ \dots \\ x_n^{(k+1)} = b_{n1}x_1^{(k+1)} + b_{n2}x_2^{(k+1)} + \dots + c_n. \end{cases} \quad (5.7.1)$$

Введем нижнюю и верхнюю треугольные матрицы

* Людвиг Зейдель (1821-1896) - немецкий астроном и математик.

$$B_1 = \begin{pmatrix} 0 & 0 & 0 & \dots & 0 \\ b_{21} & 0 & 0 & \dots & 0 \\ b_{31} & b_{32} & 0 & \dots & 0 \\ \dots & \dots & \dots & \dots & \dots \\ b_{n1} & b_{n2} & b_{n3} & \dots & 0 \end{pmatrix} \quad \text{и} \quad B_2 = \begin{pmatrix} 0 & b_{12} & b_{13} & \dots & b_{1n} \\ 0 & 0 & b_{23} & \dots & b_{2n} \\ 0 & 0 & 0 & \dots & b_{3n} \\ \dots & \dots & \dots & \dots & \dots \\ 0 & 0 & 0 & \dots & 0 \end{pmatrix}.$$

Тогда расчетная формула метода Зейделя примет такой вид

$$\bar{x}^{(k+1)} = B_1 \bar{x}^{(k+1)} + B_2 \bar{x}^{(k)} + \bar{c}. \quad (5.7.2)$$

5.8. Сходимость метода Зейделя

Теорема 5.6. Пусть выполнено условие $\|B_1\| + \|B_2\| < 1$. Тогда при любом выборе $\bar{x}^{(0)}$ метод Зейделя сходится и верна оценка погрешности

$$\|\bar{x}^{(n)} - \bar{x}_{\text{точн.}}\| \leq q^n \|\bar{x}^{(0)} - \bar{x}_{\text{точн.}}\|, \quad \text{где } q = \frac{\|B_2\|}{1 - \|B_1\|} < 1. \quad (5.8.1)$$

Это опять априорная оценка. Ее практическое использование затруднительно. Теорема 5.6 сформулирована для матриц B_1 и B_2 , однако для метода Зейделя справедлива теорема, аналогичная теореме 5.4, использующая только норму матрицы B . Апостериорная оценка погрешности метода Зейделя дается следующей теоремой.

Теорема 5.7. Если $\|B\| < 1$, то для метода Зейделя справедлива следующая апостериорная оценка погрешности:

$$\|\bar{x}^{(n)} - \bar{x}_{\text{точн.}}\| \leq \frac{\|B_2\|}{1 - \|B\|} \|\bar{x}^{(n)} - \bar{x}^{(n-1)}\|, \quad n \geq 1. \quad (5.8.2)$$

Существует устойчивое заблуждение, что метод Зейделя сходится всегда быстрее, чем метод простых итераций. Это действительно так только в случае, когда матрица A симметрична и положительно определена. Однако в общем случае возможны ситуации, когда метод Якоби сходится, а метод Зейделя или сходится медленнее, или вообще расходится. Возможны и противоположные ситуации. Дело в том, что эти методы ориентированы на решение разных классов задач: метод Якоби - на системы с матрицами, близкими к диагональным, а метод Зейделя - на системы с матрицами, близкими к нижним треугольным.

Пример. Решить систему линейных алгебраических уравнений методом простых итераций и методом Зейделя с $\varepsilon = 10^{-4}$:

$$\begin{cases} 38.1000x_1 + 0.1601x_2 + 0.1916x_3 + 0.2230x_4 = 124.0015, \\ 0.1237x_1 + 37.2000x_2 + 0.1866x_3 + 0.2180x_4 = 128.3760, \\ 0.1187x_1 + 0.1502x_2 + 36.3000x_3 + 0.2131x_4 = 132.3800, \\ 0.1137x_1 + 0.1452x_2 + 0.1766x_3 + 35.4000x_4 = 136.0134. \end{cases}$$

Дана система $A\bar{x} = \bar{b}$, преобразуем ее к виду $\bar{x} = B\bar{x} + \bar{c}$:

$$\begin{cases} x_1 = -0.0042x_2 - 0.0050x_3 - 0.0059x_4 + 3.2546, \\ x_2 = -0.0033x_1 - 0.0050x_3 - 0.0059x_4 + 3.4510, \\ x_3 = -0.0033x_1 - 0.0041x_2 - 0.0059x_4 + 3.6468, \\ x_4 = -0.0032x_1 - 0.0041x_2 - 0.0050x_3 + 3.8422. \end{cases}$$

Вычислим норму (5.1.11) для матриц B , B_1 и B_2 . Имеем

$$B = \begin{pmatrix} 0 & -0.0042 & -0.0050 & -0.0059 \\ -0.0033 & 0 & -0.0050 & -0.0059 \\ -0.0033 & -0.0041 & 0 & -0.0059 \\ -0.0032 & -0.0041 & -0.0050 & 0 \end{pmatrix}, \|B\|_\infty = \max_{1 \leq i \leq n} \sum_{j=1}^n |b_{ij}|.$$

Отсюда $\|B\|_\infty = \max(0.0033, 0.0042, 0.0050, 0.0059) = 0.0059 \ll 1$. Аналогично $\|B_1\|_\infty = 0.042$, $\|B_2\|_\infty = 0.059$. Оба метода должны сходиться очень быстро. Пусть $\bar{x}^{(0)} = (0, 0, 0, 0)^T$. Тогда по методу Якоби $\bar{x}^{(1)} = (3.2546, 3.4510, 3.6468, 3.8422)^T$. Далее

$$\begin{aligned} x_1^{(2)} &= -0.0042 \cdot 3.4510 - 0.0050 \cdot 3.6468 - 0.0059 \cdot 3.8422 + 3.2546 = 3.1992, \\ x_2^{(2)} &= -0.0033 \cdot 3.2546 - 0.0050 \cdot 3.6468 - 0.0059 \cdot 3.8422 + 3.4510 = 3.3994, \\ x_3^{(2)} &= -0.0033 \cdot 3.2446 - 0.0041 \cdot 3.4510 - 0.0059 \cdot 3.8422 + 3.6468 = 3.5992, \\ x_4^{(2)} &= -0.0032 \cdot 3.2546 - 0.0041 \cdot 3.4510 - 0.0050 \cdot 3.6468 + 3.8422 = 3.7994. \end{aligned}$$

Дальнейшие итерации проводятся аналогично:

$$\begin{aligned} x_1^{(3)} &= -0.0042 \cdot 3.3994 - 0.0050 \cdot 3.5992 - 0.0059 \cdot 3.7994 + 3.2546 = 3.1992, \\ x_2^{(3)} &= -0.0033 \cdot 3.1992 - 0.0050 \cdot 3.5992 - 0.0059 \cdot 3.7994 + 3.4510 = 3.4000, \\ x_3^{(3)} &= -0.0033 \cdot 3.1992 - 0.0041 \cdot 3.3994 - 0.0059 \cdot 3.7994 + 3.6468 = 3.5999, \\ x_4^{(3)} &= -0.0032 \cdot 3.1992 - 0.0041 \cdot 3.3994 - 0.0050 \cdot 3.5992 + 3.8422 = 3.8000. \end{aligned}$$

Последняя итерация дает

$$\begin{aligned} x_1^{(4)} &= -0.0042 \cdot 3.4000 - 0.0050 \cdot 3.5999 - 0.0059 \cdot 3.8000 + 3.2546 = 3.2000, \\ x_2^{(4)} &= -0.0033 \cdot 3.1999 - 0.0050 \cdot 3.5999 - 0.0059 \cdot 3.8000 + 3.4510 = 3.4000, \\ x_3^{(4)} &= -0.0033 \cdot 3.1999 - 0.0041 \cdot 3.4000 - 0.0059 \cdot 3.8000 + 3.6468 = 3.6000, \\ x_4^{(4)} &= -0.0032 \cdot 3.1999 - 0.0041 \cdot 3.4000 - 0.0050 \cdot 3.5999 + 3.8422 = 3.8000. \end{aligned}$$

Поскольку $\|B\| = 0.0059 \ll 1$, то $\frac{\|B\|}{1 - \|B\|} \approx 1$ и $\|\bar{x}^{(n)} - \bar{x}_{\text{точн.}}\| = \|\bar{x}^{(n)} - \bar{x}^{(n-1)}\|$. Таким образом, достигнута точность $\varepsilon = 10^{-4}$ за четыре итерации.

Вычисления по методу Зейделя для этого примера мало отличаются от выше приведенных. Действительно, $\bar{x}^{(0)} = (0, 0, 0, 0)^T$, $\bar{x}^{(1)} = (3.2546, 3.4510, 3.6468, 3.8422)^T$.

$$\begin{aligned} x_1^{(2)} &= 3.1992, \text{ то есть то же выражение, что и в предыдущем случае.} \\ x_2^{(2)} &= -0.0033 \cdot 3.1992 - 0.0050 \cdot 3.6468 - 0.0059 \cdot 3.8422 + 3.4510 = 3.3995, \\ x_3^{(2)} &= -0.0033 \cdot 3.1992 - 0.0041 \cdot 3.3995 - 0.0059 \cdot 3.8422 + 3.6468 = 3.5996, \\ x_4^{(2)} &= -0.0032 \cdot 3.1992 - 0.0041 \cdot 3.3995 - 0.0050 \cdot 3.5996 + 3.8422 = 3.8000. \end{aligned}$$

Третья итерация, оказавшаяся последней, дает

$$\begin{aligned} x_1^{(3)} &= -0.0042 \cdot 3.3995 - 0.0050 \cdot 3.5996 - 0.0059 \cdot 3.8000 + 3.2546 = 3.2000, \\ x_2^{(3)} &= -0.0033 \cdot 3.2000 - 0.0050 \cdot 3.5996 - 0.0059 \cdot 3.8000 + 3.4510 = 3.4000, \\ x_3^{(3)} &= -0.0033 \cdot 3.2000 - 0.0041 \cdot 3.4000 - 0.0059 \cdot 3.8000 + 3.6468 = 3.6000, \\ x_4^{(3)} &= -0.0032 \cdot 3.2000 - 0.0041 \cdot 3.4000 - 0.0050 \cdot 3.6000 + 3.8422 = 3.8000. \end{aligned}$$

Достигнут тот же результат за три итерации.

5.9. Лабораторная работа № 9. Решение систем линейных алгебраических уравнений методом простых итераций

Метод простых итераций используется для решения разреженных систем большой размерности ($\sim 10^4 \div 10^6$), причем матрица такой системы помимо разреженности должна быть близкой к диагональной. Метод сходится тем быстрее, чем меньше норма матрицы коэффициентов B , при этом для сходимости метода необходимо $\|B\| < 1$. Основная формула метода

$$\bar{x}^{(k+1)} = B\bar{x}^{(k)} + \bar{c}, \quad k = 0, 1, 2, \dots \quad (5.9.1)$$

Если система линейных уравнений задана в традиционной форме $A \cdot \bar{x} = \bar{b}$, ее сначала нужно привести к форме (5.9.1) методом Якоби.

Рассмотрим **пример** решения такой системы в пакете Mathcad:

$$A := \begin{pmatrix} 0.4000 & 0.0003 & 0.0008 & 0.0014 \\ -0.0029 & -0.5000 & -0.0018 & -0.0012 \\ -0.0055 & -0.0050 & -1.4000 & -0.0039 \\ -0.0082 & -0.0076 & -0.0070 & -2.3000 \end{pmatrix} \quad b := \begin{pmatrix} 0.1220 \\ -0.2532 \\ -0.9876 \\ -2.0812 \end{pmatrix}$$

$$\text{norml}(A) = 2.306$$

$$n := 4 \quad i := 1 \dots n \quad c_i := \frac{b_i}{A_{i,i}} \quad c = \begin{pmatrix} 0.305 \\ 0.5064 \\ 0.7054286 \\ 0.9048696 \end{pmatrix}$$

$$i := 1 \dots n \quad j := 1 \dots n \quad B_{i,j} := \frac{-A_{i,j}}{A_{i,i}} \quad i := 1 \dots n \quad B_{i,i} := 0$$

$$B = \begin{pmatrix} 0 & -7.5 \cdot 10^{-4} & -2 \cdot 10^{-3} & -3.5 \cdot 10^{-3} \\ -5.8 \cdot 10^{-3} & 0 & -3.6 \cdot 10^{-3} & -2.4 \cdot 10^{-3} \\ -3.9285714 \cdot 10^{-3} & -3.5714286 \cdot 10^{-3} & 0 & -2.7857143 \cdot 10^{-3} \\ -3.5652174 \cdot 10^{-3} & -3.3043478 \cdot 10^{-3} & -3.0434783 \cdot 10^{-3} & 0 \end{pmatrix}$$

$$\text{norml}(B) = 0.013.$$

Встроенная подпрограмма `norml` вычисляет первые нормы матриц A и B . Так как $\|B\| = 0.013$, то по теореме 5.4 подразд. 5.6 метод итераций должен сходиться при любом начальном приближении.

Предыдущие операторы программы приводят систему уравнений, заданную в виде $A \cdot \bar{x} = \bar{b}$, к виду $\bar{x} = B \cdot \bar{x} + \bar{c}$ по формулам подразд. 5.5. Сам процесс последовательных приближений можно записать в векторно-матричной форме всего лишь одной строкой программы:

$$xx^{(1)} := c \quad k := 2 \dots 11 \quad xx^{(k)} := c + B \cdot xx^{(k-1)}$$

Здесь в качестве начального приближения выбран вектор \bar{c} ; видно, что из десяти заданных итераций для достижения заданной точности 10^{-7} потребовалось лишь три.

	1	2	3	4	5
1	0.305	0.3000423	0.3000754	0.300075	0.300075
2	0.5064	0.4999198	0.4999802	0.4999797	0.4999797
3	0.7054286	0.6999011	0.6999574	0.6999569	0.6999569
4	0.9048696	0.8999619	0.9000178	0.9000173	0.9000173

$xx^{(k)}$ - k -й столбец матрицы xx размерности (4×11) , где хранятся все приближения к точному решению.

Для проверки решим эту же систему $A \cdot \bar{x} = \bar{b}$ встроенной программой **lsolve**, которую мы уже использовали в лабораторной работе № 4.

$$zz := \text{lsolve}(A, b) \quad zz = \begin{pmatrix} 0.3000750 \\ 0.4999797 \\ 0.6999569 \\ 0.9000173 \end{pmatrix}$$

В заключение приведем две подпрограммы, реализующие приведение исходной системы линейных уравнений $A \cdot \bar{x} = \bar{b}$ к нужной форме и итерационные вычисления по методу Якоби:

$$\begin{aligned} \text{itera}(A, n) := & \left| \begin{array}{l} \text{for } i \in 1..n \\ \quad a \leftarrow \frac{1}{A_{i,i}} \\ \quad \text{for } j \in 1..n \\ \quad \quad B_{i,j} \leftarrow -A_{i,j} \cdot a \\ \quad \quad B_{i,j} \leftarrow 0 \text{ if } i=j \\ \quad \text{B} \end{array} \right| \\ & \text{Jakobi}(A, b, \varepsilon) := \left| \begin{array}{l} n \leftarrow \text{rows}(A) \\ B \leftarrow \text{itera}(A, n) \\ \text{for } i \in 1..n \\ \quad c_i \leftarrow \frac{b_i}{A_{i,i}} \\ \quad x_i \leftarrow 0 \\ \quad x1_i \leftarrow c_i \\ \quad \text{norm} \leftarrow \text{norm1}(B) \\ \quad \text{return norm if norm} \geq 1 \\ \quad \text{while norm} > \varepsilon \\ \quad \quad \text{for } i \in 1..n \\ \quad \quad \quad x_i \leftarrow c_i + \left[\sum_{j=1}^n (B_{i,j} \cdot x_j) \right] \\ \quad \quad \quad \text{norm} \leftarrow \sqrt{x \cdot x} \\ \quad \quad \quad x1 \leftarrow x - x \\ \quad \quad \quad \text{norm} \leftarrow \frac{\sqrt{x \cdot x}}{\text{norm}} \\ \quad \quad \quad x \leftarrow x \\ \quad \quad \quad \text{x} \end{array} \right| \end{aligned}$$

$$\varepsilon := 10^{-7} \quad z := \text{Jakobi}(A, b, \varepsilon)$$

$$z = \begin{pmatrix} 0.3000750 \\ 0.4999797 \\ 0.6999569 \\ 0.9000173 \end{pmatrix}$$

Задание № 1. Методом простых итераций с точностью $\varepsilon = 10^{-7}$ решить систему линейных алгебраических уравнений, заданную в форме $A \cdot \bar{x} = \bar{b}$:

№ варианта	Матрица A				Вектор правой части \bar{b}
	1.70	0.03	0.04	0.05	
1	0.00	0.80	0.01	0.02	0.4803
	-0.03	-0.02	-0.10	0.00	-0.0802
	-0.05	-0.04	-0.03	-1.00	-1.0007
	3.00	0.38	0.49	0.59	1.5136
2	0.11	2.10	0.32	0.43	1.4782
	-0.05	0.05	1.20	0.26	1.0830
	-0.22	-0.11	-0.01	0.30	0.3280

№ варианта	Матрица A				Вектор правой части \bar{b}
	0.77	0.04	-0.21	0.18	
3	-0.45	1.23	-0.06	0.00	-0.8800
	-0.26	-0.34	1.11	0.00	-0.6200
	-0.05	0.26	-0.34	1.12	-1.1700
	0.79	-0.12	0.34	0.16	-0.6400
4	-0.34	1.08	-0.17	0.18	1.4200
	-0.16	-0.34	0.85	0.31	-0.4200
	-0.12	0.26	0.08	0.75	0.8300
	0.99	-0.02	0.62	-0.08	-1.3000
5	-0.03	0.72	-0.33	0.07	1.1000
	-0.09	-0.13	0.58	-0.28	-1.7000
	-0.19	0.23	-0.08	0.63	1.5000
	3.68	0.16	0.18	0.22	1.1604
6	0.12	3.59	0.18	0.21	1.2025
	0.11	0.14	3.50	0.21	1.2409
	0.11	0.14	0.17	3.11	1.2757
	3.55	0.15	0.18	0.21	1.0834
7	0.11	3.46	0.16	0.19	1.1239
	0.12	0.14	3.37	0.20	1.1607
	0.10	0.13	0.17	3.28	1.1938
	2.38	0.10	0.12	0.14	5.0897
8	0.08	2.29	0.11	0.14	5.3487
	0.07	0.09	2.20	0.15	5.5712
	0.06	0.08	0.11	1.10	5.7570
	1.00	-0.17	0.33	-0.18	-1.2000
9	0.00	0.82	-0.43	0.08	0.3300
	-0.22	-0.18	0.79	-0.07	0.4800
	-0.08	-0.07	-0.21	0.96	-1.2000
	0.68	0.18	-0.02	-0.21	1.8300
10	-0.16	0.88	0.14	-0.27	-0.6500
	-0.37	-0.27	1.02	0.24	2.2300
	-0.12	-0.21	0.18	0.75	-1.1300
	0.58	0.32	-0.03	0.00	0.4400
11	-0.11	1.26	0.36	0.00	1.4200

	-0.12	-0.08	1.14	0.24	-0.8300
	-0.15	0.35	0.18	1.00	-1.4200
12	0.82	0.34	0.12	-0.15	-1.3300
	-0.11	0.77	0.15	-0.32	0.8400
	-0.05	0.12	0.86	0.18	-1.1600
	-0.12	-0.08	-0.06	1.00	0.5700
13	0.87	-0.23	0.44	0.05	2.3000
	-0.24	1.00	0.31	-0.15	-0.1800
	-0.06	-0.15	1.00	0.23	1.4400
	-0.72	0.08	0.05	1.00	2.4200
14	0.85	-0.05	0.08	-0.14	-0.4800
	-0.32	1.13	0.12	-0.11	1.2400
	-0.17	-0.06	1.08	-0.12	1.1500
	-0.21	0.16	-0.36	1.00	-0.8800

№ варианта	Матрица A				Вектор правой части \bar{b}
	0.97	0.05	-0.22	0.33	
15	0.97	0.05	-0.22	0.33	0.4300
	-0.22	0.45	0.08	-0.07	-1.8000
	-0.33	-0.13	1.08	0.05	-0.8000
	-0.08	-0.17	-0.29	0.67	1.7000
16	4.30	0.22	0.27	0.32	2.6632
	0.10	3.40	0.21	0.26	2.7779
	0.04	0.09	2.50	0.20	2.5330
	-0.03	0.03	0.08	1.60	1.9285
17	5.60	0.27	0.33	0.39	4.0316
	0.15	4.70	0.27	0.33	4.3135
	0.09	0.15	3.80	0.27	4.2353
	0.03	0.09	0.15	2.90	3.7969
18	6.90	0.32	0.39	0.46	5.6632
	0.19	6.00	0.33	0.41	6.1119
	0.13	0.21	5.10	0.35	6.2000
	0.08	0.15	0.22	4.20	5.9275
19	8.20	0.37	0.45	0.53	7.5591
	0.23	7.30	0.39	0.48	8.1741
	0.18	0.26	6.40	0.42	8.4281
	0.12	0.21	0.29	5.50	8.3210
20	9.50	0.42	0.51	0.60	9.7191
	0.28	8.60	0.46	0.55	10.5000
	0.22	0.32	7.70	0.50	10.9195
	0.17	0.26	0.35	6.80	10.9775
21	0.87	-0.22	0.33	-0.07	0.1100
	0.00	0.55	0.23	-0.07	-0.3300
	-0.11	0.00	1.08	-0.18	0.8500
	-0.08	-0.09	-0.33	0.79	-1.7000
22	0.68	0.16	0.08	-0.15	2.4200
	-0.16	1.23	-0.11	0.21	1.4300
	-0.05	0.08	1.00	-0.34	-0.1600
	-0.12	-0.14	0.18	0.94	1.6200
23	1.00	-0.08	0.23	-0.32	1.3400
	-0.16	1.23	-0.18	-0.16	-2.3300

	-0.15	-0.12	0.68	0.18	0.3400
	-0.25	-0.21	0.16	0.97	0.6300
24	10.80	0.05	0.06	0.07	12.1430
	0.03	9.90	0.05	0.06	13.0897
	0.04	0.04	9.00	0.08	13.6744
	0.02	0.03	0.04	8.10	13.8972
25	12.10	5.28	0.64	0.75	14.8310
	0.37	11.20	5.86	0.69	15.9430
	0.31	0.42	10.30	6.44	16.6926
	2.60	0.37	4.81	19.40	17.0800
26	13.40	5.81	0.70	0.82	17.7828
	0.41	12.50	6.50	0.77	19.0599
	0.36	0.48	11.60	7.18	19.9744
	0.31	0.43	0.54	10.70	20.5261

№ варианта	Матрица A				Вектор правой части \bar{b}
	0.94	-0.18	-0.33	-0.16	
27	-0.32	1.00	-0.23	0.05	-1.1200
	-0.16	0.08	1.00	0.12	0.4300
	-0.09	-0.22	0.13	1.00	0.8300
	1.00	-0.34	-0.23	0.06	1.4200
28	-0.11	1.23	0.18	-0.36	-0.6600
	-0.23	0.12	0.84	0.35	1.0800
	-0.12	-0.12	0.47	0.82	1.7200
	0.68	0.23	-0.11	0.06	0.6700
29	-0.18	0.88	0.33	0.00	-0.8800
	-0.12	-0.32	1.05	-0.07	0.1800
	-0.05	0.11	-0.09	1.12	1.4400
	0.77	0.14	-0.06	0.12	1.2100
30	-0.12	1.00	-0.32	0.18	-0.7200
	-0.08	0.12	0.77	-0.32	-0.5800
	-0.25	-0.22	-0.14	1.00	1.5600

5.10. Постановка задачи нахождения собственных чисел

Вычисление собственных чисел и векторов матриц - одна из весьма сложных вычислительных задач. Рассмотрим несколько методов их нахождения.

Число λ называется собственным значением (собственным числом) матрицы A , если существует ненулевой вектор \bar{x} , удовлетворяющий уравнению

$$A\bar{x} = \lambda \bar{x}. \quad (5.10.1)$$

\bar{x} называется собственным вектором матрицы A , соответствующим собственному числу λ .

Перепишем (5.10.1) в виде $(A - \lambda E)\bar{x} = 0$. Это однородная система уравнений. Чтобы существовало решение этой системы $\bar{x} \neq 0$, необходимо чтобы определитель матрицы этой системы был равен нулю, то есть

$$\det(A - \lambda E) = 0. \quad (5.10.2)$$

Раскрытие этого уравнения приводит к характеристическому уравнению

$$\lambda^n + p_1\lambda^{n-1} + p_2\lambda^{n-2} + \dots + p_{n-1}\lambda + p_n = 0, \quad (5.10.3)$$

представляющему собой алгебраическое уравнение степени n . Решение уравнения (5.10.3) дает все n собственных значений $\lambda_1, \lambda_2, \dots, \lambda_n$ квадратной матрицы $A_{n \times n}$. Если матрица A симметрическая, то все ее собственные значения - вещественные числа.

Классические численные методы решения проблемы собственных значений сводились, в конечном счете, к решению характеристического уравнения. Методы такого класса получили название прямых методов. Однако если размерность матрицы A будет составлять несколько десятков, а то и сотен единиц, такой подход становится неудовлетворительным. Для матриц высокого порядка возникает трудно разрешимая проблема вычисления коэффициентов характеристического уравнения с достаточной точностью.

Современные методы решения проблемы собственных значений основаны на совершенно иных идеях. Как правило, это итерационные алгоритмы, в которых собственные значения вычисляются как пределы некоторых числовых последовательностей без предварительного определения коэффициентов характеристического уравнения. Одновременно с нахождением собственных значений определяются и собственные векторы.

5.11. Подобные матрицы

Матрицы A и B называются подобными, если $B = P^{-1}AP$. При этом должна существовать невырожденная матрица подобия P .

Преобразование подобия есть результат перехода к новому базису в пространстве n -мерных векторов. Пусть $\bar{y} = A\bar{x}$. Произведем замену переменных $\bar{x} = Px'$, $\bar{y} = Py'$. Тогда равенство $\bar{y} = A\bar{x}$ примет вид $\bar{y}' = P^{-1}AP\bar{x}'$, то есть то же преобразование в новых переменных осуществляется не матрицей A , а подобной ей матрицей $P^{-1}AP$. Рассмотрим характеристическое уравнение матрицы $P^{-1}AP$:

$$\det(P^{-1}AP - \lambda E) = \det[P^{-1}(A - \lambda E)P] = \det P^{-1} \det(A - \lambda E) \det P = \det(A - \lambda E).$$

Таким образом, характеристические уравнения, а следовательно, и собственные числа матриц A и $P^{-1}AP$ совпадают. Соответствующие собственные векторы не совпадают, они связаны равенством $\bar{x} = Px'$.

Оказывается, существуют преобразования подобия, приводящие исходную матрицу A к треугольному виду. Именно, справедлива

Теорема 5.8. Любую квадратную матрицу A с помощью преобразования подобия можно привести к следующему виду:

$$P^{-1}AP = \Lambda = \begin{pmatrix} \lambda_1 & \sigma_1 & 0 & 0 & \dots & 0 & 0 \\ 0 & \lambda_2 & \sigma_2 & 0 & \dots & 0 & 0 \\ 0 & 0 & \lambda_3 & \sigma_3 & \dots & 0 & 0 \\ \dots & \dots & \dots & \dots & \dots & \dots & \dots \\ 0 & 0 & 0 & 0 & \dots & \lambda_{n-1} & \sigma_{n-1} \\ 0 & 0 & 0 & 0 & \dots & 0 & \lambda_n \end{pmatrix}. \quad (5.11.1)$$

Здесь $\lambda_1, \lambda_2, \dots, \lambda_n$ - собственные числа матрицы A . Если $\lambda_i = \lambda_{i+1}$, то $\sigma_i = 1$, если же $\lambda_i \neq \lambda_{i+1}$, то $\sigma_i = 0$. Матрица (5.11.1) называется жордановой* формой матрицы A .

5.12. Локализация собственных значений

Иногда удается получить грубые оценки расположения собственных чисел. Имеется несколько локализационных теорем, одна из них - теорема Гершгорина.** Пусть $r_i = \sum_{\substack{j=1 \\ j \neq i}}^n |a_{ij}|$ -

* Мари Энмон Камиль Жордан (1838-1922) - французский математик.

сумма вне диагональных элементов i -й строки матрицы A . Обозначим через S_i круг радиуса r_i на комплексной плоскости с центром в точке a_{ii} ; круги S_i называются кругами Гершгорина.

Теорема 5.9. Все собственные значения матрицы A лежат в объединении кругов S_1, S_2, \dots, S_n . Если какой-либо круг Гершгорина изолирован, то он содержит ровно одно собственное значение матрицы A .

Действительно, запишем уравнение $(A - \lambda E)\bar{x} = 0$ в скалярной форме: $a_{ii}x_i + \sum_{i \neq j}^n a_{ij}x_j - \lambda x_i = 0$, где x_i - максимальная по модулю координата вектора \bar{x} .

$$(a_{ii} - \lambda)x_i = -\sum_{i \neq j}^n a_{ij}x_j. \text{ Так как } \left| \frac{x_j}{x_i} \right| \leq 1, \text{ то } |a_{ii} - \lambda| \leq \sum_{i \neq j}^n |a_{ij}| \left| \frac{x_j}{x_i} \right| \leq r_i, \text{ то есть } \lambda \in S_i.$$

Пример. Найти все собственные числа матрицы A и изобразить на плоскости круги Гершгорина:

$$A = \begin{pmatrix} 1 & 2 & 1 \\ 9 & 1 & 2 \\ 2 & -1 & 0 \end{pmatrix}. \text{ Вычислим сначала радиусы кругов. } r_1 = 3, r_2 = 11, r_3 = 3.$$

Итак, $S_1 = \{z \in C : |z - 1| \leq 3\}$, $S_2 = \{z \in C : |z - 1| \leq 11\}$, $S_3 = \{z \in C : |z| \leq 3\}$. Очевидно, S_1 и $S_3 \subset S_2$ и объединением всех кругов является S_2 . Все собственные числа лежат внутри

круга S_2 . Найдем $\lambda_1, \lambda_2, \lambda_3$, решив характеристическое уравнение для данной матрицы A :

$$\begin{vmatrix} 1 - \lambda & 2 & 1 \\ 9 & 1 - \lambda & 2 \\ 2 & -1 & -\lambda \end{vmatrix} = 0 = \lambda^3 - 2\lambda^2 - 17\lambda + 1. \text{ Для}$$

решения кубического уравнения применим формулы Кардано.* Уравнение вида $ax^3 + bx^2 + cx + d = 0$ заменой переменной $y = x + \frac{b}{3a}$ приводится к виду

$$y^3 + 3py + 2q = 0, \text{ где } 2q = \frac{2b^3}{27a^3} - \frac{bc}{3a^2} + \frac{d}{a},$$

$$3p = \frac{c}{a} - \frac{b^2}{3a^2}. \quad \text{В нашем случае}$$

$2q = -10.925925$, $q = -5.462963$, $3p = -18.333333$, $p = -6.111111$. Число действительных решений уравнения $y^3 + 3py + 2q = 0$ зависит от знака дискриминанта $D = q^2 + p^3$. $D = -198.38 < 0$. Если $D < 0$, то уравнение имеет три действительных различных корня, которые находятся следующим образом. Вычисляются

$r = \operatorname{sgn} q \cdot \sqrt{|p|} = -2.4720661$, $\cos \varphi = \frac{q}{r^3} = 0.361616$, $\varphi = 68^\circ.8005$, $\frac{\varphi}{3} = 22^\circ.9335$. Далее находятся сами корни: $y_1 = -2r \cos \frac{\varphi}{3} = 4.5533381$, $y_2 = 2r \cos \left(60^\circ - \frac{\varphi}{3}\right) = -3.945106$,

** Семен Аронович Гершгорин (1901 - 1933) - российский математик.

* Джироламо Кардано (1501-1576) - итальянский математик.

$$y_3 = 2r \cos\left(60^\circ + \frac{\varphi}{3}\right) = -0.608232, \quad x_1 = y_1 - \frac{b}{3a} = y_1 + \frac{2}{3} = 5.241808, \quad x_2 = -3.278439,$$

$x_3 = 0.058435$. Итак, $\lambda_1 = 5.242$, $\lambda_2 = -3.278$, $\lambda_3 = 0.058$ с точностью $\varepsilon = 10^{-3}$.

При вычислении собственных чисел и собственных векторов симметрических матриц важную роль играет функция

$$\rho(x) = \frac{(Ax, \bar{x})}{(\bar{x}, \bar{x})}, \quad (5.12.1)$$

которая называется отношением Релея. **

Теорема 5.10. Пусть A - симметрическая матрица. Тогда

1) минимальное и максимальное собственное число матрицы A равны

$$\lambda_{\min} = \min_{x \neq 0} \rho(x), \quad \lambda_{\max} = \max_{x \neq 0} \rho(x); \quad (5.12.2)$$

2) вектор \bar{x} является стационарной точкой $\rho(x)$, то есть $\text{grad}\rho(x) = 0$ тогда и только тогда, когда \bar{x} - собственный вектор матрицы A .

5.13. Степенной метод

Пусть необходимо вычислить максимальное по модулю собственное число λ , матрицы A , причем $|\lambda_1| > |\lambda_2| > \dots > |\lambda_n|$. В степенном методе используется следующая последовательность формул. Выбирается произвольный начальный вектор $\bar{x}^{(0)}$ и строится последовательность векторов $\{\bar{x}^{(k)}\}_{k=0}^{\infty}$ и приближений к собственному числу λ_1 $\{\lambda_1^{(k)}\}_{k=0}^{\infty}$ таким образом:

$$\begin{cases} \bar{x}^{(k)} = A\bar{x}^{(k-1)}, \\ \lambda_1^{(k)} = \frac{(\bar{x}^{(k)}, \bar{x}^{(k)})}{(\bar{x}^{(k-1)}, \bar{x}^{(k-1)})}. \end{cases} \quad (5.13.1)$$

Правая часть формулы для $\lambda_1^{(k)}$ в (5.13.1) - это отношение Релея при $\bar{x} = \bar{x}^{(k-1)}$. Так как $\bar{x}^{(k)} = A\bar{x}^{(k-1)}$, то $\lambda_1^{(k)} = \frac{(A\bar{x}^{(k-1)}, \bar{x}^{(k-1)})}{(\bar{x}^{(k-1)}, \bar{x}^{(k-1)})} = \rho(\bar{x}^{(k-1)})$.

Теорема 5.11. Пусть в разложении $\bar{x}^{(0)} = c_1 \bar{e}_1 + c_2 \bar{e}_2 + \dots + c_n \bar{e}_n$ по базису $\{\bar{e}_i\}_{i=1}^n$ из собственных векторов произвольной матрицы A , собственные числа которой удовлетворяют условию $|\lambda_1| > |\lambda_2| > \dots > |\lambda_n|$ и $c_1 \neq 0$. Тогда $\lambda_1^{(k)} \rightarrow \lambda_1$ при $k \rightarrow \infty$ и справедлива следующая оценка погрешности:

$$\delta(\lambda_1^{(k)}) = \frac{|\lambda_1^{(k)} - \lambda_1|}{|\lambda_1|} \leq c \left| \frac{\lambda_2}{\lambda_1} \right|^k, \quad c = \text{const}. \quad (5.13.2)$$

При практических вычислениях вектор $\bar{x}^{(k)}$ нормируют, чтобы не было переполнения или исчезновений порядка. Формулы (5.13.1) тогда приобретают вид

** Джон Уильям Рэлей (1842 -1919) - английский физик и математик.

$$\begin{cases} \bar{y}^{(k)} = A\bar{x}^{(k-1)}, \\ \lambda^{(k)} = \frac{\bar{y}^{(k)}}{\bar{x}^{(k)}}, \\ \bar{x}^{(k)} = \frac{\bar{y}^{(k)}}{\|\bar{y}^{(k)}\|}, \|\bar{x}^{(0)}\| = 1. \end{cases} \quad (5.13.3)$$

В общем случае при решении проблемы собственных значений не существует эффективных апостериорных оценок погрешностей. Для симметрических матриц можно использовать следующий результат.

Теорема 5.12. Пусть λ^* - произвольное число, а \bar{x}^* - произвольный ненулевой вектор. Тогда для любой симметрической матрицы A существует собственное число λ такое, что справедлива оценка

$$|\lambda - \lambda^*| \leq \frac{\|A\bar{x}^* - \lambda^*\bar{x}^*\|}{\|\bar{x}^*\|}. \quad (5.13.4)$$

В частности для степенного метода, если $\bar{x}^{(k-1)}$ - приближенно вычисленный собственный вектор, а $\lambda_1^{(k)}$ - приближенное значение собственного числа, то

$$\begin{cases} |\lambda_1 - \lambda_1^{(k)}| \leq \frac{\|\bar{x}^{(k)} - \lambda_1^{(k)}\bar{x}^{(k-1)}\|}{\|\bar{x}^{(k-1)}\|}, \\ \bar{x}^{(k)} = A\bar{x}^{(k-1)}. \end{cases} \quad (5.13.5)$$

Пример. Для матрицы из предыдущего примера вычислить λ_1 и \bar{x}_1 , взяв в качестве начального приближения $\bar{x}^{(0)} = (1, 0, 0)^T$.

Вычисления будем производить по формулам (5.13.3): $\bar{x}^{(0)} = \begin{pmatrix} 1 \\ 0 \\ 0 \end{pmatrix}$, $\bar{y}^{(0)} = A\bar{x}^{(0)} = \begin{pmatrix} 1 \\ 9 \\ 2 \end{pmatrix}$

$$= \begin{pmatrix} 1 & 2 & 1 \\ 9 & 1 & 2 \\ 2 & -1 & 0 \end{pmatrix} \cdot \begin{pmatrix} 1 \\ 0 \\ 0 \end{pmatrix} = \begin{pmatrix} 1 \\ 9 \\ 2 \end{pmatrix}, \lambda_1 = \left(\bar{y}^{(1)}, \bar{x}^{(0)} \right) = (1 \cdot 1 + 0 \cdot 9 + 0 \cdot 2) = 1, \|\bar{y}^{(1)}\| = \sqrt{1^2 + 9^2 + 2^2} = 9.273618, \bar{x}^{(1)} = \frac{\bar{y}^{(1)}}{\|\bar{y}^{(1)}\|} = \begin{pmatrix} 0.107833 \\ 0.970495 \\ 0.215666 \end{pmatrix}.$$

Результаты вычислений шестнадцати итераций приведены в следующей таблице:

Номер итерации k	$\lambda_1^{(k)}$	$x_1^{(k)}$	$x_2^{(k)}$	$x_3^{(k)}$
0	-	1.0000000	0.000000	0.000000
1	1	0.107833	0.970945	0.215666
2	2.401193	0.670351	0.707602	-0.223450
3	5.560346	0.282393	0.954482	0.096011
4	4.128699	0.524958	0.846417	-0.089436

5	5.375470	0.320293	0.811308	-0.051321
6	4.703964	0.461342	0.886644	-0.032067
7	5.425668	0.404845	0.914362	0.006624
8	5.085983	0.439969	0.897778	-0.020557
9	5.298928	0.417814	0.908526	-0.002034
10	5.171134	0.431704	0.901906	-0.014094
11	5.251758	0.422957	0.906120	-0.007328
12	5.226316	0.426272	0.904522	-0.011520
13	5.216012	0.426340	0.904508	-0.009965
14	5.220049	0.426316	0.904519	-0.009929
15	5.219940	0.426331	0.904513	-0.009940
16	5.220039	0.426322	0.904517	-0.009933

Видно, что при вычислении по формуле Кардано не достигнута даже точность $\varepsilon = 10^{-3}$, так как $\lambda_1 \approx 5.220$, а не 5.242, как в предыдущем примере. Главный недостаток этого метода - медленная сходимость, пропорциональная $|\lambda_2/\lambda_1|$.

Для вычисления следующего собственного числа можно использовать сдвиг на $\sigma = \lambda_1^*$. После сдвига рассматривается матрица $\tilde{A} = A - \sigma E$, собственными числами которой являются числа $\lambda_i - \sigma$. В этом случае число $\lambda_1 - \lambda_1^*$ будет минимальным по модулю, следовательно, степенной метод даст другое число, например, $\lambda_2 - \lambda_1^*$, сдвиг которого максимальен.

5.14. Вычисление собственных векторов методом обратных итераций

Если найдено достаточно точное приближение λ_j^* к собственному числу λ_j , то, казалось бы, можно вычислить \bar{e}_j из уравнения по определению

$$(A - \lambda_j^* E) \bar{e}_j = 0. \quad (5.14.1)$$

Однако из-за приближенности λ_j^* матрица $A - \lambda_j^* E$ будет плохо обусловленной, но невырожденной и, следовательно, $\bar{e}_j = 0$. Таким образом, уравнение (5.14.1) мало подходит для определения \bar{e}_j .

В методе обратных итераций приближения к собственному вектору определяются как последовательные решения системы уравнений

$$\begin{cases} (A - \lambda_j^* E) \bar{y}^{(k+1)} = \bar{x}^{(k)}, \\ \bar{x}^{(k+1)} = \frac{\bar{y}^{(k+1)}}{\|\bar{y}^{(k+1)}\|}, \end{cases} \quad (5.14.2)$$

причем в качестве $\bar{x}^{(0)}$ берут любой ненулевой вектор, например, $\bar{x}^{(0)} = (1, 1, \dots, 1)^T$.

Рассмотрим $\bar{x}^{(0)}$ и $\bar{y}^{(1)}$ в виде разложения по базису из собственных векторов $\bar{x}^{(0)} = \sum_{i=1}^n c_i \bar{e}_i$, $\bar{y}^{(1)} = \sum_{i=1}^n \alpha_i \bar{e}_i$. Тогда $(A - \lambda_j^* E) \bar{y}^{(1)} = (A - \lambda_j^* E) \sum_{i=1}^n \alpha_i \bar{e}_i = \sum_{i=1}^n \alpha_i (\lambda_i - \lambda_j^*) \bar{e}_i$, так

как в базисе $\{\bar{e}_i\}_{i=1}^n$ $A = \begin{pmatrix} \lambda_1 & 0 & \dots & 0 \\ 0 & \lambda_2 & \dots & 0 \\ \dots & \dots & \dots & \dots \\ 0 & 0 & \dots & \lambda_n \end{pmatrix}$. Система (5.14.2) преобразуется к виду

$\sum_{i=1}^n \alpha_i (\lambda_i - \lambda_j^*) \bar{e}_i = \sum_{i=1}^n c_i \bar{e}_i$, то есть $\alpha_i (\lambda_i - \lambda_j^*) = c_i$, $\alpha_i = \frac{c_i}{\lambda_i - \lambda_j^*}$. Отсюда $\bar{y}^{(1)} = \sum_{i=1}^n \alpha_i \bar{e}_i = \sum_{i=1}^n \frac{c_i}{\lambda_i - \lambda_j^*} \bar{e}_i = \frac{1}{\lambda_j - \lambda_j^*} \left(c_j \bar{e}_j + \sum_{i \neq j} \frac{\lambda_j - \lambda_j^*}{\lambda_i - \lambda_j^*} c_i \bar{e}_i \right)$. Если λ_j^* достаточно близко к λ_j , то $|\lambda_j - \lambda_j^*| \ll |\lambda_i - \lambda_j^*|$ для всех $i \neq j$ и второе слагаемое в скобке, то есть $\sum_{i \neq j} \frac{\lambda_j - \lambda_j^*}{\lambda_i - \lambda_j^*} c_i \bar{e}_i$, будет мало. Тогда $\bar{y}^{(1)} \approx \frac{c_j}{\lambda_j - \lambda_j^*} \bar{e}_j = k \bar{e}_j$, то есть векторы $\bar{y}^{(1)}$ и \bar{e}_j будут почти коллинеарными. Тогда вектор $\bar{x}^{(k)}$ будет сходиться к \bar{e}_j по направлению со скоростью геометрической прогрессии со знаменателем $q = \max_{i \neq j} \left| \frac{\lambda_j - \lambda_j^*}{\lambda_i - \lambda_j^*} \right|$. Чаще всего $q \ll 1$ и метод сходится очень быстро.

Практически метод обратных итераций используют вместе с отношением Релея:

$$\begin{cases} \lambda^{(k)} = \rho(\bar{x}^{(k)}) = \left(\bar{A} \bar{x}^{(k)}, \bar{x}^{(k)} \right), k \geq 0, \\ (\bar{A} - \lambda^{(k)} \bar{E}) \bar{y}^{(k+1)} = \bar{x}^{(k)}, \\ \bar{x}^{(k+1)} = \frac{\bar{y}^{(k+1)}}{\|\bar{y}^{(k+1)}\|}, \bar{x}^{(0)} = 1. \end{cases} \quad (5.14.3)$$

Пример. Найдем \bar{e}_1 для матрицы из предыдущего примера, положив $\bar{x}^{(0)} = (1, 1, 1)^T$ и приняв $\lambda_1 = 5.220$.

Запишем систему из (5.14.3) при $k = 0$:

$$\begin{cases} -4.220y_1 + 2y_2 + y_3 = 1, \\ 9y_1 - 4.220y_2 + 2y_3 = 1, \\ 2y_1 - y_2 - 5.220y_3 = 1. \end{cases}$$

Решим ее

методом Гаусса с выбором главного элемента:

$$\begin{cases} 9.000000y_1 - 4.220000y_2 + 2.000000y_3 = 1.000000, \\ 0.021289y_2 + 1.937778y_3 = 1.468889, \\ -0.062222y_2 - 5.664444y_3 = 0.777778. \end{cases}$$

$$\begin{cases} 9.000000y_1 - 4.220000y_2 + 2.000000y_3 = 1.000000, \\ -0.062222y_2 - 5.664444y_3 = 0.777778, \\ 3.097 \cdot 10^{-6}y_2 = 1.734963. \end{cases}$$

Отсюда $y_2 = 560207.58$, $y_3 = -6153.83$, $y_1 = 264043.38$. Наконец, нормируя вектор $\bar{y}^{(1)}$, полу

лучим $\bar{e}_1 = \bar{x}_1 = \begin{pmatrix} 0.426327 \\ 0.904514 \\ -0.009936 \end{pmatrix}$.

5.15. Лабораторная работа № 10. Вычисление собственных значений (чисел) и векторов матриц

К настоящему времени имеется много различных алгоритмов решения проблемы собственных значений. Все они эффективны, но достаточно сложны и трудоемки. К наиболее простым методам нахождения собственных чисел относится степенной метод. К сожалению, им можно определить лишь два максимальных по модулю собственных значения.

Также довольно простой метод обратных итераций, очень похожий на степенной по алгоритму, позволяет находить собственный вектор, соответствующий заданному собственному числу. Это итеративный метод, он дает приближение к собственному вектору, сходящееся к нему по направлению. Следует заметить, что при вычислении собственных векторов имеется определенный произвол. Собственные векторы линейного оператора, отвечающие определенному собственному числу, образуют линейное пространство. Различные способы вычисления собственных векторов дают разные базисы этого пространства и, следовательно, разные координаты собственных векторов, удовлетворяющих, однако, основному определению собственного вектора $\bar{A}\bar{x} = \lambda_i \bar{x}$.

В среде Mathcad собственные значения и векторы находятся функциями **eigenvals**, **eigenvec**, **eigenvecs**, **genvals** и **genvecs**. Рассмотрим их все по порядку.

Встроенная функция **eigenvals(A)** вычисляет вектор собственных значений матрицы A , функция **genvals** – вектор обобщенных собственных значений λ_i матрицы A , удовлетворяющий условию

$$\bar{A}\bar{x} = \lambda_i \bar{N}\bar{x}, \quad (5.15.1)$$

где N – матрица с действительными элементами. Если задать $N = E$, то результаты работы этих двух подпрограмм будут одинаковыми.

Введем с клавиатуры начало программы

$$\begin{aligned} \text{ORIGIN} := 1 & \quad A := \begin{pmatrix} 1 & 1 & 2 & 3 \\ 0 & 2 & 2 & 4 \\ 0 & 0 & 1 & -2 \\ 0 & 0 & 0 & 2 \end{pmatrix} \quad TOL = 10^{-3} \quad E := \text{identity}(4) \quad E = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} \\ c := \text{eigenvals}(A) \quad c = \begin{pmatrix} 1 \\ 2 \\ 1 \\ 2 \end{pmatrix} & \quad c1 := \text{genvals}(A, E) \quad c1 = \begin{pmatrix} 1 \\ 2 \\ 1 \\ 2 \end{pmatrix}. \end{aligned}$$

Аналогично вычисляются и собственные векторы, соответствующие заданным собственным числам:

$$v := \text{eigenvec}(A, c_1) \quad v = \begin{pmatrix} -0.643 \\ -0.685 \\ 0.343 \\ 0 \end{pmatrix} \quad v1 := \text{eigenvec}(A, c_2) \quad v1 = \begin{pmatrix} 0.643 \\ 0.737 \\ -0.187 \\ 0.093 \end{pmatrix}$$

Эти вектора удовлетворяют основному определению $\bar{A}\bar{x} = \lambda_i \bar{x}$ (см. ниже). Функции **eigenvecs** и **genvecs** также дают все нормированные собственные векторы заданной матрицы. Координаты собственных векторов расположены по столбцам результирующей матрицы, порядок расположения собственных векторов соответствует порядку собственных чисел,

возвращаемых функциями **eigenvals** и **genvals**. Однако в данном случае собственные векторы относятся к иному базису линейного пространства и имеют другие координаты.

$$\begin{aligned}
A \cdot v &= \begin{pmatrix} -0.643 \\ -0.685 \\ 0.343 \\ 0 \end{pmatrix} & c_1 \cdot v &= \begin{pmatrix} -0.643 \\ -0.685 \\ 0.343 \\ 0 \end{pmatrix} & A \cdot v1 &= \begin{pmatrix} 1.287 \\ 1.473 \\ -0.373 \\ 0.187 \end{pmatrix} & A \cdot v &= \begin{pmatrix} 1.287 \\ 1.473 \\ -0.373 \\ 0.187 \end{pmatrix} \\
vv := \text{eigenvects}(A) \quad vv &= \begin{pmatrix} 1 & 0.707 & 0 & 0.408 \\ 0 & 0.707 & 0.894 & 0 \\ 0 & 0 & -0.447 & 0.816 \\ 0 & 0 & 0 & -0.408 \end{pmatrix} & vm := \text{genvecs}(A, E) \\
vm &= \begin{pmatrix} 1 & 0.707 & 0 & -0.408 \\ 0 & 0.707 & -0.894 & 0 \\ 0 & 0 & 0.447 & -0.816 \\ 0 & 0 & 0 & 0.408 \end{pmatrix} & A \cdot vv &= \begin{pmatrix} 1 & 1.414 & 0 & 0.816 \\ 0 & 1.414 & 0.894 & 0 \\ 0 & 0 & -0.447 & 1.633 \\ 0 & 0 & 0 & -0.816 \end{pmatrix} \\
c_1 \cdot vv^{(1)} &= \begin{pmatrix} 1 \\ 0 \\ 0 \\ 0 \end{pmatrix} & c_2 \cdot vv^{(2)} &= \begin{pmatrix} 1.414 \\ 1.414 \\ 0 \\ 0 \end{pmatrix} & c_3 \cdot vv^{(3)} &= \begin{pmatrix} 0 \\ 0.894 \\ -0.447 \\ 0 \end{pmatrix} & c_4 \cdot vv^{(4)} &= \begin{pmatrix} 0.816 \\ 0 \\ 1.633 \\ -0.816 \end{pmatrix}
\end{aligned}$$

Противоположные знаки компонент векторов $vv^{(3)}$ и $vv^{(4)}$ в матрицах vv и vm объясняются противоположной направленностью собственных векторов, получаемых подпрограммами `eigenvecs` и `genvecs`.

Поскольку функции, реализующей формулы (5.13.1) степенного метода, в пакете Mathcad нет, составим такую подпрограмму сами. В качестве исходных данных будем рассматривать саму матрицу A и произвольный вектор \bar{x}_0 , необходимый для запуска процесса итераций по формулам (5.13.1). Сама подпрограмма весьма проста и имеет следующий вид:

```

l(x0,A) := | n←rows(A)
             | norm1←| x0 |
             | x←x0
             | kol←0
             | while norm1>TOL
             |   y←A·x
             |   l1←y·x
             |   norm←| y |
             |   for iε 1..n
             |     yi← $\frac{y_i}{norm}$ 
             |     norm1←| y - x |
             |     x←y
             |     kol←kol + 1
             |   break if kol>100

```

Найдем два первых максимальных по модулю собственных числа матрицы

$$x0 := \begin{pmatrix} 1 \\ 1 \\ 1 \\ 1 \end{pmatrix} \quad \lambda_1 := \lambda(x0, A) \quad \lambda_1 = 2 \quad A1 := A - \lambda_1 \cdot E \quad \lambda_2 := \lambda(x0, A1) + \lambda_1 \quad \lambda_2 = 1$$

Вычисление собственного вектора можно встроить в подпрограмму $\lambda(x0, A)$ по формулам (5.14.2) или (5.14.3). Запрограммируем, однако, его нахождение отдельно:

```

    while norm > TOL
        y ← lsolve(A1, x0)
        norm ← |y|
        for i ∈ 1..n
            y_i ← y_i / norm
            x0_i ← |y_i| - |x0_i|
        norm ← √|x0|
        x0 ← y
    x0

```

$$v1 := \text{evect}(\lambda_1, A) \quad v1 = \begin{pmatrix} 0.632 \\ 0.738 \\ -0.211 \\ 0.105 \end{pmatrix} \quad A \cdot v1 = \begin{pmatrix} 1.265 \\ 1.476 \\ -0.422 \\ 0.211 \end{pmatrix} \quad v1 \cdot \lambda_1 = \begin{pmatrix} 1.265 \\ 1.476 \\ -0.422 \\ 0.211 \end{pmatrix}$$

Задание № 1. Определить собственные значения и собственные вектора матрицы A средствами пакета Mathcad, затем найти максимальное по модулю собственное число и соответствующий ему собственный вектор с помощью подпрограмм $\lambda(x0, A)$ и $\text{evect}(\lambda_1, A)$.

$$1. \quad A = \begin{pmatrix} 1 & 1.5 & 2.5 & 3.5 \\ 1.5 & 1 & 2 & 1.6 \\ 2.5 & 2 & 1 & 1.7 \\ 3.5 & 1.6 & 1.7 & 1 \end{pmatrix}$$

$$2. \quad A = \begin{pmatrix} 1 & 1.2 & 2 & 0.5 \\ 1.2 & 1 & 0.4 & 1.2 \\ 2 & 0.4 & 2 & 1.5 \\ 0.5 & 1.2 & 1.5 & 1 \end{pmatrix}$$

$$3. \quad A = \begin{pmatrix} 1 & 1.2 & 2 & 0.5 \\ 1.2 & 1 & 0.5 & 1 \\ 2 & 0.5 & 2 & 1.5 \\ 0.5 & 1 & 1.5 & 0.5 \end{pmatrix}$$

$$4. \quad A = \begin{pmatrix} 2.5 & 1 & -0.5 & 2 \\ 1 & 2 & 1.2 & 0.4 \\ -0.5 & 1.2 & -1 & 1.5 \\ 2 & 0.4 & 1.5 & 1 \end{pmatrix}$$

$$5. A = \begin{pmatrix} 2 & 1 & 1.4 & 0.5 \\ 1 & 1 & 0.5 & 1 \\ 1.4 & 0.5 & 2 & 1.2 \\ 0.5 & 1 & 1.2 & 0.5 \end{pmatrix}$$

$$7. A = \begin{pmatrix} 2 & 1.5 & 3.5 & 4.5 \\ 1.5 & 2 & 2 & 1.6 \\ 3.5 & 2 & 2 & 1.7 \\ 4.5 & 1.6 & 1.7 & 2 \end{pmatrix}$$

$$9. A = \begin{pmatrix} 1.2 & 0.5 & 2 & 1 \\ 0.5 & 1 & 0.8 & 2 \\ 2 & 0.8 & 1 & 1 \\ 1 & 2 & 1 & 2 \end{pmatrix}$$

$$11. A = \begin{pmatrix} 1.2 & 0.5 & 2 & 1 \\ 0.5 & 1 & 0.6 & 2 \\ 2 & 0.6 & 1 & 1 \\ 1 & 2 & 1 & 2 \end{pmatrix}$$

$$13. A = \begin{pmatrix} 1.6 & 1 & 1.4 & 1 \\ 1 & 1 & 0.5 & 2 \\ 1.4 & 0.5 & 2 & 1.2 \\ 1 & 2 & 1.2 & 0.5 \end{pmatrix}$$

$$15. A = \begin{pmatrix} 0.5 & 1.2 & 2 & 1 \\ 1.2 & 2 & 0.5 & 1.2 \\ 2 & 0.5 & 1 & 0.5 \\ 1 & 1.2 & 0.5 & 1.6 \end{pmatrix}$$

$$17. A = \begin{pmatrix} 1 & 1.5 & 1.2 & 0.5 \\ 1.5 & 2 & 0.4 & 2 \\ 1.2 & 0.4 & 1.5 & 1.4 \\ 0.5 & 2 & 1.4 & 1.3 \end{pmatrix}$$

$$19. A = \begin{pmatrix} 1 & 1.5 & 0.4 & 2 \\ 1.5 & -1.2 & 1 & -0.5 \\ 0.4 & 1 & 2 & 1.2 \\ 2 & -0.5 & 1.2 & 2.5 \end{pmatrix}$$

$$21. A = \begin{pmatrix} 1.5 & 1.6 & 1.7 & 1.8 \\ 1.6 & 2.5 & 1.2 & 1.3 \\ 1.7 & 1.2 & 3.5 & 1.4 \\ 1.8 & 1.3 & 1.4 & 4.6 \end{pmatrix}$$

$$23. A = \begin{pmatrix} 1.6 & 1.6 & 1.7 & 1.8 \\ 1.6 & 2.6 & 1.3 & 1.3 \\ 1.7 & 1.3 & 3.6 & 1.4 \\ 1.8 & 1.3 & 1.4 & 4.6 \end{pmatrix}$$

$$6. A = \begin{pmatrix} 2 & 1.2 & -1 & 1 \\ 1.2 & 0.5 & 2 & -1 \\ -1 & 2 & -1.5 & 0.2 \\ 1 & -1 & 0.2 & 1.5 \end{pmatrix}$$

$$8. A = \begin{pmatrix} 1 & 0.5 & 1.2 & -1 \\ 0.5 & 2 & -0.5 & 0 \\ 1.2 & -0.5 & -1 & 1.4 \\ -1 & 0 & 1.4 & 1 \end{pmatrix}$$

$$10. A = \begin{pmatrix} 0.5 & 1.2 & 1 & 0.9 \\ 1.2 & 2 & 0.5 & 1.2 \\ 1 & 0.5 & 1 & 1 \\ 0.5 & 1.2 & 1 & 2.2 \end{pmatrix}$$

$$12. A = \begin{pmatrix} 2 & 1.5 & 4.5 & 5.5 \\ 1.5 & 3 & 2 & 1.6 \\ 4.5 & 2 & 3 & 1.7 \\ 5.5 & 1.6 & 1.7 & 3 \end{pmatrix}$$

$$14. A = \begin{pmatrix} 2.4 & 0.5 & 2 & 1 \\ 0.5 & 1 & 0.8 & 2 \\ 2 & 0.8 & 1 & 0.5 \\ 1 & 2 & 0.5 & 1.2 \end{pmatrix}$$

$$16. A = \begin{pmatrix} 1.8 & 1.6 & 1.7 & 1.8 \\ 1.6 & 2.8 & 1.5 & 1.3 \\ 1.7 & 1.5 & 3.8 & 1.4 \\ 1.8 & 1.3 & 1.4 & 4.8 \end{pmatrix}$$

$$18. A = \begin{pmatrix} 1 & 0.5 & -0.5 & 1 \\ 0.5 & -1 & 2 & 0 \\ -0.5 & 2 & 1 & -1.5 \\ 1 & 0 & -1.5 & 2 \end{pmatrix}$$

$$20. A = \begin{pmatrix} 1.9 & 1.6 & 1.7 & 1.8 \\ 1.6 & 2.9 & 1.6 & 1.3 \\ 1.7 & 1.6 & 3.9 & 1.4 \\ 1.8 & 1.3 & 1.4 & 4.9 \end{pmatrix}$$

$$22. A = \begin{pmatrix} 0.5 & 1 & 1.2 & 2 \\ 1 & 1.2 & -0.5 & 0.6 \\ 1.2 & -0.5 & 1 & -1 \\ 3 & 0.6 & -1 & 1.2 \end{pmatrix}$$

$$24. A = \begin{pmatrix} 2 & 1.6 & 1.7 & 1.8 \\ 1.6 & 3 & 1.7 & 1.3 \\ 1.7 & 1.7 & 4 & 1.4 \\ 1.8 & 1.3 & 1.4 & 5 \end{pmatrix}$$

$$25. A = \begin{pmatrix} 0.5 & 1.4 & 2 & 1 \\ 1.4 & 1 & 0 & 1.5 \\ 2 & 0 & 2.5 & 2 \\ 1 & 1.5 & 2 & 1 \end{pmatrix}$$

$$27. A = \begin{pmatrix} 1.7 & 1.6 & 1.7 & 1.8 \\ 1.6 & 2.7 & 1.4 & 1.3 \\ 1.7 & 1.4 & 3.7 & 1.4 \\ 1.8 & 1.3 & 1.4 & 4.7 \end{pmatrix}$$

$$29. A = \begin{pmatrix} 1.6 & 0.4 & 1 & 2 \\ 0.4 & 1 & 0.5 & 1 \\ 1 & 0.5 & 0 & 0.2 \\ 2 & 1 & 0.2 & 0.5 \end{pmatrix}$$

$$26. A = \begin{pmatrix} 1 & 1.2 & 0.3 & 2 \\ 1.2 & 0.5 & 1 & 0.7 \\ 0.3 & 1 & -0.4 & 1 \\ 2 & 0.7 & 1 & 1.5 \end{pmatrix}$$

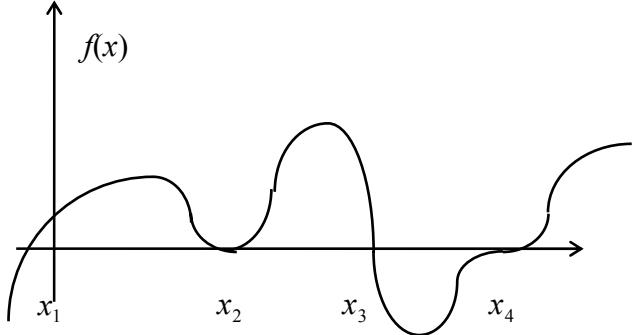
$$28. A = \begin{pmatrix} 2 & 1.7 & 1.6 & 4.5 \\ 1.7 & 2 & 2 & 3.5 \\ 1.6 & 2 & 1 & 1.5 \\ 4.5 & 3.5 & 1.5 & 1 \end{pmatrix}$$

$$30. A = \begin{pmatrix} 3 & 1.7 & 1.6 & 5.5 \\ 1.7 & 1 & 2 & 4.5 \\ 1.6 & 2 & 3 & 1.5 \\ 5.5 & 4.5 & 1.5 & 1 \end{pmatrix}$$

6. ЧИСЛЕННЫЕ МЕТОДЫ РЕШЕНИЯ НЕЛИНЕЙНЫХ УРАВНЕНИЙ И СИСТЕМ

6.1. Решение нелинейных уравнений

Пусть дано уравнение $f(x) = 0$. Корнем этого уравнения называется такое значение x_k , при котором $f(x_k) = 0$. Корень x_k называется простым, если $f'(x_k) \neq 0$, в противном случае - кратным. Целое число m называется кратностью корня x_k , если $f^{(k)}(x_k) = 0$, для $k = 1, 2, \dots, m-1$, а $f^{(m)}(x_k) \neq 0$. Геометрически корень x_k соответствует точке пересечения графика функции $y = f(x)$ с осью Ox . Корень кратный, когда пересечение происходит под нулевым углом. На рисунке x_1 , x_3 - простые корни, x_2 , x_4 - кратные.



В подавляющем большинстве случаев представить решение уравнения $f(x) = 0$ в виде конечной замкнутой формулы оказывается невозможным. Даже для простейшего алгебраического уравнения n -й степени $x^n + a_{n-1}x^{n-1} + \dots + a_1x + a_0 = 0$ явные формулы корней известны для $n = 2, 3, 4$. Уже для

уравнения пятой (и более высоких степеней) таких формул не существует.

Задача отыскания корней нелинейного уравнения решается в два этапа. Первый называется этапом локализации (отделения) корней, второй - этапом итерационного уточнения корней. Отрезок $[a, b]$, содержащий только один корень x_k уравнения $f(x) = 0$, называется отрезком локализации корня x_k . Способы локализации корней многообразны, и указать универсальный метод не представляется возможным. Иногда отрезок локализации известен либо он определяется из физических соображений. В простых случаях хороший результат может дать графический метод. На этапе итерационного уточнения корней с точностью ϵ используют тот или иной итерационный метод, позволяющий строить последовательность $x^{(0)}, x^{(1)}, \dots, x^{(n)}$ приближений к корню x_T . Итерационный метод называют одношаговым, если для вычисления очередного приближения $x^{(n+1)}$ используется только одно предыдущее приближение $x^{(n)}$ и k -шаговым, если для вычисления $x^{(n+1)}$ используется k предыдущих приближений $x^{(n-k+1)}, x^{(n-k+2)}, \dots, x^{(n)}$. Столько же данных необходимо для начального приближения, чтобы запустить метод.

Скорость сходимости - одна из важнейших характеристик итерационных методов. Говорят, что метод сходится со скоростью геометрической прогрессии, знаменатель которой $q < 1$, если для $\forall n \in N$ справедлива оценка:

$$|x^{(n)} - x_{\text{точн.}}| \leq c_0 q^n, \text{ где } c_0 = \text{const}. \quad (6.1.1)$$

При определении скорости сходимости метода используют понятие порядка сходимости. Если справедливо неравенство

$$|x^{(n+1)} - x_{\text{точн.}}| \leq c |x^{(n)} - x_{\text{точн.}}|^p, \text{ где } c > 0, p \geq 1, \quad (6.1.2)$$

то число p называют порядком сходимости. Если $p = 1$, то сходимость линейная (сходимость геометрической прогрессии), при $p > 1$ сходимость называется сверхлинейной. Если $p = 2$, скорость сходимости называют квадратичной.

Если $p = 1$, то есть метод обладает линейной сходимостью, можно установить справедливость формулы

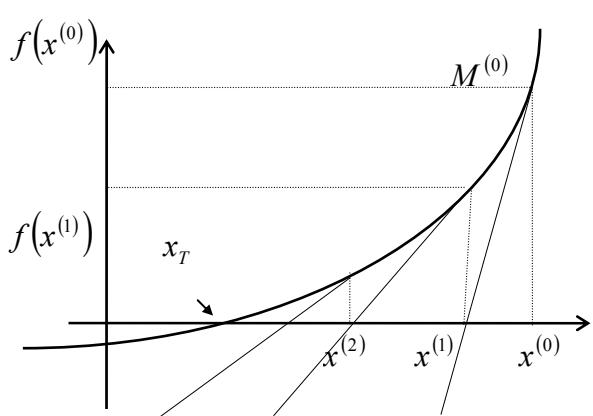
$$|x^{(n)} - x_{\text{точн.}}| \leq q^n |x^{(0)} - x_{\text{точн.}}|, \quad n \geq 0; \quad (6.1.3)$$

смотрите метод простых итераций или метод Зейделя в предыдущей главе. Если же $p > 1$, то справедлива оценка

$$|x^{(n)} - x_{\text{точн.}}| \leq c_1 (q^p)^n, \quad n \geq 0, \quad \text{где } c_1 = \tilde{c} |x^{(0)} - x_{\text{точн.}}|. \quad (6.1.4)$$

6.2. Метод Ньютона для уравнений

Этот знаменитый метод - один из наиболее эффективных методов решения самых различных нелинейных задач. Выведем его расчетную формулу, исходя из простейших геометрических соображений. Соответствующая иллюстрация приведена на рисунке слева. Пусть



$x^{(0)}$ - заданное начальное условие. В точке $M^{(0)}$ с координатами $M^{(0)}(x^{(0)}, f(x^{(0)}))$ проведем касательную к графику функции $y = f(x)$ и за новое приближение $x^{(1)}$ примем абсциссу точки пересечения этой касательной с осью Ox . Далее поступают аналогично, за приближение $x^{(2)}$ берут абсциссу точки пересечения с осью Ox касательной, проведенной к графику в точке $M^{(1)}(x^{(1)}, f(x^{(1)}))$ и так далее. Уравнение касательной, проведенной к графику $y = f(x)$ в точке $(x^{(n)}, f(x^{(n)}))$, имеет вид $y = f(x^{(n)}) + f'(x^{(n)})(x - x^{(n)})$. Если положить здесь $y = 0$, тогда абсцисса точки $x^{(n+1)}$ пересечения касательной с осью Ox будет удовлетворять этому уравнению, то есть $0 = f(x^{(n)}) + f'(x^{(n)})(x^{(n+1)} - x^{(n)})$. Отсюда

$$x^{(n+1)} = x^{(n)} - \frac{f(x^{(n)})}{f'(x^{(n)})}, \quad n \geq 0, \quad f'(x^{(n)}) \neq 0. \quad (6.2.1)$$

Это и есть основная рабочая формула метода Ньютона или метода касательных, получившего свое второе название благодаря указанной геометрической интерпретации.

6.3. Сходимость метода Ньютона и трудности его применения

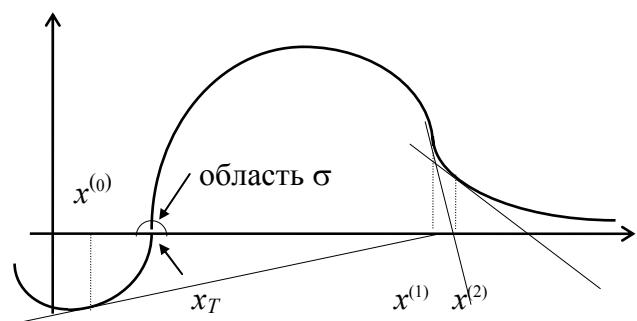
Теорема 6.1. Пусть x_k - простой корень уравнения $f(x) = 0$, в некоторой окрестности которого функция $y = f(x)$ дважды непрерывно дифференцируема. Тогда найдется такая малая σ -окрестность корня x_T , что при произвольном выборе начального приближения $x^{(0)}$ из этой окрестности итерационная последовательность метода Ньютона не выходит за пределы этой окрестности и справедлива оценка

$$|x^{(n+1)} - x_T| \leq c |x^{(n)} - x_T|^2, \quad n \geq 0, \quad (6.3.1)$$

где $c = 1/\sigma$. Это означает, что метод Ньютона сходится с квадратичной скоростью. Грубо говоря, на каждой итерации число верных знаков приближения примерно удваивается.

Простота и высокая скорость сходимости делает метод Ньютона чрезвычайно привлекательным. Однако имеются две существенные трудности. Первая из них - необходимость вычисления производной $f'(x)$.

Это часто либо невозможно сделать, либо вычисление $f'(x)$ оказывается слишком дорогим. В таких ситуациях прибегают к различным модификациям метода Ньютона. Вторая трудность - его локальная сходимость. Это значит, что последовательные



приближения сходятся к точному решению лишь в малой σ -окрестности точки x_T . Неудачный выбор начального приближения может дать расходящуюся последовательность (см. рисунок слева). Для преодоления этой трудности часто на практике используют метод Ньютона в сочетании с каким-нибудь медленно, но гарантированно сходящимся методом типа метода бисекций.

6.4. Метод Ньютона решения систем нелинейных уравнений

Решение систем нелинейных уравнений - задача существенно более сложная, чем уравнения типа $f(x) = 0$. Пусть дана система n нелинейных уравнений с n неизвестными вида

$$\begin{cases} f_1(x_1, x_2, \dots, x_n) = 0, \\ f_2(x_1, x_2, \dots, x_n) = 0, \\ \dots \\ f_n(x_1, x_2, \dots, x_n) = 0. \end{cases} \quad (6.4.1)$$

На практике, к сожалению, эта задача встречается гораздо чаще предыдущей. Как и в случае уравнения с одним неизвестным, отыскание решения начинают с этапа локализации. Для каждого из искомых решений \bar{x} указывают множество, которое содержит только одно это решение и расположено в достаточно малой его окрестности. Часто в качестве такого множества выступает параллелепипед или шар в n -мерном пространстве. Сама задача локализации в большинстве случаев очень сложна, иногда полное решение задачи локализации невозможно. В простейших случаях могут быть использованы графические методы.

В методе Ньютона применяется линеаризация системы (6.4.1). Пусть по выбранному $\bar{x}^{(0)}$ построены приближения $\bar{x}^{(1)}, \bar{x}^{(2)}, \dots, \bar{x}^{(k)}$. Заменим в системе (6.4.1) каждую из функций $f_i(x), i = 1, 2, \dots, n$ линейной частью ее разложения по формуле Тейлора в точке $\bar{x}^{(k)}$:

$$f_i(\bar{x}) \approx f_i(\bar{x}^{(k)}) + \sum_{j=1}^n \frac{\partial f_i}{\partial x_j}(\bar{x}^{(k)})(x_j - x_j^{(k)}). \quad (6.4.2)$$

Тогда вместо системы (6.4.1) получим

$$\begin{cases} f_1(\bar{x}^{(k)}) + \sum_{j=1}^n \frac{\partial f_1}{\partial x_j}(\bar{x}^{(k)})(x_j - x_j^{(k)}) = 0, \\ f_2(\bar{x}^{(k)}) + \sum_{j=1}^n \frac{\partial f_2}{\partial x_j}(\bar{x}^{(k)})(x_j - x_j^{(k)}) = 0, \\ \dots \\ f_n(\bar{x}^{(k)}) + \sum_{j=1}^n \frac{\partial f_n}{\partial x_j}(\bar{x}^{(k)})(x_j - x_j^{(k)}) = 0. \end{cases} \quad (6.4.3)$$

В матричной форме система (6.4.3) будет иметь вид

$$\bar{f}(\bar{x}^{(k)}) + \Phi(\bar{x}^{(k)})(\bar{x} - \bar{x}^{(k)}) = 0, \quad (6.4.4)$$

где матрица Якоби

$$\Phi(\bar{x}) = \begin{pmatrix} \frac{\partial f_1}{\partial x_1}(\bar{x}) & \frac{\partial f_1}{\partial x_2}(\bar{x}) & \dots & \frac{\partial f_1}{\partial x_n}(\bar{x}) \\ \frac{\partial f_2}{\partial x_1}(\bar{x}) & \frac{\partial f_2}{\partial x_2}(\bar{x}) & \dots & \frac{\partial f_2}{\partial x_n}(\bar{x}) \\ \dots & \dots & \dots & \dots \\ \frac{\partial f_n}{\partial x_1}(\bar{x}) & \frac{\partial f_n}{\partial x_2}(\bar{x}) & \dots & \frac{\partial f_n}{\partial x_n}(\bar{x}) \end{pmatrix}. \quad (6.4.5)$$

Если матрица $\Phi(\bar{x})$ - невырожденная, то есть $\det \Phi(\bar{x}) \neq 0$, то существует обратная матрица $\Phi^{-1}(\bar{x})$. Тогда система (6.4.4) имеет единственное решение, которое и принимается за очередное приближение $\bar{x}^{(k+1)}$ к решению \bar{x} . Таким образом, приближение $\bar{x}^{(k+1)}$ удовлетворяет равенству

$$\bar{f}(\bar{x}^{(k)}) + \Phi(\bar{x}^{(k)}) (\bar{x}^{(k+1)} - \bar{x}^{(k)}) = 0. \quad (6.4.6)$$

Из (6.4.6), как и в случае одного уравнения, легко выводится итерационная формула метода Ньютона для систем нелинейных уравнений:

$$\bar{x}^{(k+1)} = \bar{x}^{(k)} - \Phi^{-1}(\bar{x}^{(k)}) \bar{f}(\bar{x}^{(k)}). \quad (6.4.7)$$

Формула (6.4.7) редко используется для непосредственного вычисления $\bar{x}^{(k+1)}$ в силу своей трудоемкости из-за необходимости обращать матрицу Якоби. Вместо этого часто решается система линейных алгебраических уравнений вида

$$\Phi(\bar{x}^{(k)}) \Delta x^{(k+1)} = -\bar{f}(\bar{x}^{(k)}) \text{ где } \Delta x^{(k+1)} = \bar{x}^{(k+1)} - \bar{x}^{(k)}. \quad (6.4.8.)$$

Очередное приближение легко находится по формуле

$$\bar{x}^{(k+1)} = \bar{x}^{(k)} + \Delta x^{(k+1)}. \quad (6.4.9)$$

6.5. Сходимость метода Ньютона для систем нелинейных уравнений

Теорема 6.2. Пусть в некоторой окрестности решения \bar{x} системы (6.4.1) функции $f_i(\bar{x})$, $i = 1, 2, \dots, n$ дважды непрерывно дифференцируемы и матрица Якоби $\Phi(\bar{x})$ невырожденная. Тогда найдется такая малая σ -окрестность решения \bar{x} , что при произвольном выборе начального приближения $\bar{x}^{(0)}$ из этой σ -окрестности итерационная последовательность метода Ньютона сходится к решению \bar{x} , не выходя за пределы σ -окрестности и справедлива оценка

$$\|\bar{x}^{(k+1)} - \bar{x}\| \leq c \|\bar{x}^{(k)} - \bar{x}\|^2, \quad k \geq 0, \quad c = \frac{1}{\sigma}. \quad (6.5.1)$$

По приведенной в начале главы классификации метод Ньютона сходится с квадратичной скоростью. Практически для окончания расчетов используют критерий

$$\|\bar{x}^{(k+1)} - \bar{x}^{(k)}\| < \varepsilon. \quad (6.5.2)$$

Все трудности метода, указанные для случая одного уравнения, сохраняются и здесь. Обостряется проблема нахождения хорошего начального приближения. Кроме того, вместо одной производной приходится вычислять n^2 частных производных, что само по себе может оказаться весьма сложным делом. В целом решение многомерной задачи значительно сложнее решения одного нелинейного уравнения.

6.6. Модификации метода Ньютона

Существует большое число модификаций метода Ньютона, позволяющих снизить его трудоемкость. Вот некоторые из них.

а. Упрощенный метод Ньютона.

В нем матрица $\Phi(\bar{x}^{(k)})$, вычисляемая заново на каждом приближении, заменяется постоянной матрицей $A = \Phi(\bar{x}^{(0)})$. В результате формулы метода приобретают следующий вид:

$$\begin{cases} A \Delta x^{(k+1)} = -\bar{f}(\bar{x}^{(k)}) \\ \bar{x}^{(k+1)} = \bar{x}^{(k)} + \Delta x^{(k+1)} \end{cases} \quad (6.6.1)$$

Этот метод обладает линейной сходимостью, причем знаменатель прогрессии q тем меньше, чем ближе $\bar{x}^{(0)}$ к \bar{x} . Число итераций здесь существенно возрастает, однако вычислительные затраты могут оказаться меньше из-за того, что вычисление матрицы Якоби происходит

дит всего один раз, кроме того в системе (6.6.1) линейные уравнения решаются с фиксированной матрицей A и различными правыми частями. Для таких систем существуют специальные экономичные способы их решения.

б. Использование формул численного дифференцирования.

Часто вычисление производных $\frac{\partial f_i}{\partial x_j}$ невозможно. В подобных ситуациях используют различные конечно-разностные аппроксимации первых частных производных. Например,

$$\frac{\partial f_i(\bar{x}^{(k)})}{\partial x_j} \approx J_{ij}^{(k)} = \frac{1}{h_j^{(k)}} [f_i(x_1^{(k)}, \dots, x_{j-1}^{(k)}, x_j^{(k)} + h_j^{(k)}, x_{j+1}^{(k)}, \dots, x_n^{(k)}) - f_i(x_1^{(k)}, \dots, x_n^{(k)})]. \quad (6.6.2)$$

Формулы метода в этом случае примут вид

$$\begin{cases} J^{(k)} \Delta \bar{x}^{(k+1)} = -\bar{f}(\bar{x}^{(k)}), \\ \bar{x}^{(k+1)} = \bar{x}^{(k)} + \Delta \bar{x}^{(k+1)}. \end{cases} \quad (6.6.3)$$

В этом методе проблему представляет выбор шагов $h_j^{(k)}$. Это также не простая задача. От выбора $h_j^{(k)}$ зависит точность приближения $J^{(k)}$ к $\Phi(\bar{x}^{(k)})$, а следовательно, и скорость сходимости.

в. Метод секущих.

В этом методе $\bar{h}^{(k)} = \bar{x}^{(k-1)} - \bar{x}^{(k)}$. Формулы (6.6.2) и (6.6.3) остаются без изменения. Однако, для того чтобы начать вычисления, необходимо задать два начальных приближения $\bar{x}^{(0)}$ и $\bar{x}^{(1)}$. Если они выбраны удачно, то метод сходится со сверхлинейной сходимостью с порядком сходимости $p = \frac{\sqrt{5} + 1}{2}$.

Пример. Решить систему нелинейных уравнений методом Ньютона с $\varepsilon = 10^{-3}$:

$$\begin{cases} \sin(x+y) - 1.1x = 0.1, \\ x^2 + y^2 = 1. \end{cases}$$

Здесь две переменные: $x_1 = x$ и $x_2 = y$. Вычислим матрицу Якоби в аналитической форме:

$$\begin{cases} f_1(x, y) = \sin(x+y) - 1.1x - 0.1, \\ f_2(x, y) = x^2 + y^2 - 1. \end{cases} \Rightarrow \begin{cases} \frac{\partial f_1}{\partial x} = \cos(x+y) - 1.1, \\ \frac{\partial f_1}{\partial y} = \cos(x+y), \\ \frac{\partial f_2}{\partial x} = 2x, \\ \frac{\partial f_2}{\partial y} = 2y. \end{cases}$$

Итак, $\Phi(\bar{x}) = \begin{pmatrix} \frac{\partial f_1(x, y)}{\partial x} & \frac{\partial f_1(x, y)}{\partial y} \\ \frac{\partial f_2(x, y)}{\partial x} & \frac{\partial f_2(x, y)}{\partial y} \end{pmatrix} = \begin{pmatrix} \cos(x+y) - 1.1 & \cos(x+y) \\ 2x & 2y \end{pmatrix}$. Пусть $\bar{x}^{(0)} = \begin{pmatrix} 0.1 \\ 1.0 \end{pmatrix}$, $\bar{x}^{(0)} \neq 0$,

так как могут быть случаи деления на нуль. Основная формула метода $\bar{x}^{(k+1)} = \bar{x}^{(k)} - \Phi^{-1}(\bar{x}^{(k)}) \bar{f}(\bar{x}^{(k)})$ поэтому на каждом шаге необходимо знать $\bar{f}(\bar{x})$ и $\Phi(\bar{x})$

Первый шаг. $\bar{f}(\bar{x}^{(0)}) = \begin{pmatrix} f_1(\bar{x}^{(0)}) \\ f_2(\bar{x}^{(0)}) \end{pmatrix} = \begin{pmatrix} \sin(1.1) - 0.1 \\ 1.01 - 1 \end{pmatrix} = \begin{pmatrix} 0.7912 \\ 0.0100 \end{pmatrix}$. В данном примере матрица

$\Phi(\bar{x})$ имеет столь простую структуру, что ее можно легко обратить аналитически, а не делать этого для численной матрицы на каждом шаге. Действительно,

$$\det \Phi = 2x \cos(x+y) - 2.2x - 2x \cos(x+y) = -2.2x,$$

$$\Phi^{-1}(\bar{x}) = \frac{1}{\det \Phi} \begin{pmatrix} A_{11} & A_{21} \\ A_{12} & A_{22} \end{pmatrix} = -\frac{1}{2.2x} \begin{pmatrix} 2x & -\cos(x+y) \\ -2x & \cos(x+y) - 1.1 \end{pmatrix} = \begin{pmatrix} -\frac{1}{1.1} & \frac{\cos(x+y)}{2.2x} \\ \frac{1}{1.1} & \frac{1.1 - \cos(x+y)}{2.2x} \end{pmatrix}.$$

Отсюда $\Phi^{-1}(\bar{x}^{(0)}) = \begin{pmatrix} -0.9091 & 2.0618 \\ 0.9091 & 2.9382 \end{pmatrix}$. $\bar{x}^{(1)} = \bar{x}^{(0)} - \Phi^{-1}(\bar{x}^{(0)}) \cdot \bar{f}(\bar{x}^{(0)}) =$

$$= \begin{pmatrix} 0.1 \\ 1.0 \end{pmatrix} - \begin{pmatrix} -0.9091 & 2.0618 \\ 0.9091 & 2.9382 \end{pmatrix} \cdot \begin{pmatrix} 0.7912 \\ 0.0100 \end{pmatrix} = \begin{pmatrix} 0.1 \\ 1.0 \end{pmatrix} - \begin{pmatrix} -0.6987 \\ 0.7487 \end{pmatrix} = \begin{pmatrix} 0.7987 \\ 0.2513 \end{pmatrix}.$$

Второй шаг. $\bar{f}(\bar{x}^{(1)}) = \begin{pmatrix} -0.1111 \\ -0.2989 \end{pmatrix}$, $\Phi^{-1}(\bar{x}^{(1)}) = \begin{pmatrix} -0.9091 & 0.2832 \\ 0.9091 & 0.3428 \end{pmatrix}$. Тогда

$$\bar{x}^{(2)} = \bar{x}^{(1)} - \Phi^{-1}(\bar{x}^{(1)}) \cdot \bar{f}(\bar{x}^{(1)}) =$$

$$= \begin{pmatrix} 0.7987 \\ 0.2513 \end{pmatrix} - \begin{pmatrix} -0.9091 & 0.2832 \\ 0.9091 & 0.3428 \end{pmatrix} \cdot \begin{pmatrix} -0.1111 \\ -0.2989 \end{pmatrix} = \begin{pmatrix} 0.7987 \\ 0.2513 \end{pmatrix} - \begin{pmatrix} 0.0164 \\ -0.2035 \end{pmatrix} = \begin{pmatrix} 0.7823 \\ 0.4548 \end{pmatrix}.$$

Третий шаг. $\bar{f}(\bar{x}^{(2)}) = \begin{pmatrix} -0.0157 \\ -0.1813 \end{pmatrix}$, $\Phi^{-1}(\bar{x}^{(2)}) = \begin{pmatrix} -0.9091 & 0.1903 \\ 0.9091 & 0.4488 \end{pmatrix}$.

$$\bar{x}^{(3)} = \bar{x}^{(2)} - \Phi^{-1}(\bar{x}^{(2)}) \cdot \bar{f}(\bar{x}^{(2)}) =$$

$$= \begin{pmatrix} 0.7823 \\ 0.4548 \end{pmatrix} - \begin{pmatrix} -0.9091 & 0.1903 \\ 0.9091 & 0.4488 \end{pmatrix} \cdot \begin{pmatrix} -0.0157 \\ -0.1813 \end{pmatrix} = \begin{pmatrix} 0.7823 \\ 0.4548 \end{pmatrix} - \begin{pmatrix} -0.0202 \\ -0.0956 \end{pmatrix} = \begin{pmatrix} 0.8025 \\ 0.5504 \end{pmatrix}.$$

Четвертый шаг. $\bar{f}(\bar{x}^{(3)}) = \begin{pmatrix} -0.0064 \\ -0.0531 \end{pmatrix}$, $\Phi^{-1}(\bar{x}^{(3)}) = \begin{pmatrix} -0.9091 & 0.1224 \\ 0.9091 & 0.5006 \end{pmatrix}$.

$$\bar{x}^{(4)} = \bar{x}^{(3)} - \Phi^{-1}(\bar{x}^{(3)}) \cdot \bar{f}(\bar{x}^{(3)}) =$$

$$= \begin{pmatrix} 0.8025 \\ 0.5504 \end{pmatrix} - \begin{pmatrix} -0.9091 & 0.1224 \\ 0.9091 & 0.5006 \end{pmatrix} \cdot \begin{pmatrix} -0.0064 \\ -0.0531 \end{pmatrix} = \begin{pmatrix} 0.8025 \\ 0.5504 \end{pmatrix} - \begin{pmatrix} -0.0123 \\ -0.0324 \end{pmatrix} = \begin{pmatrix} 0.8148 \\ 0.5828 \end{pmatrix}.$$

Пятый шаг. $\bar{f}(\bar{x}^{(4)}) = \begin{pmatrix} -0.0112 \\ 0.0036 \end{pmatrix}$, $\Phi^{-1}(\bar{x}^{(4)}) = \begin{pmatrix} -0.9091 & 0.0961 \\ 0.9091 & 0.5175 \end{pmatrix}$.

$$\bar{x}^{(5)} = \bar{x}^{(4)} - \Phi^{-1}(\bar{x}^{(4)}) \cdot \bar{f}(\bar{x}^{(4)}) =$$

$$= \begin{pmatrix} 0.8148 \\ 0.5828 \end{pmatrix} - \begin{pmatrix} -0.9091 & 0.0961 \\ 0.9091 & 0.5175 \end{pmatrix} \cdot \begin{pmatrix} -0.0112 \\ 0.0036 \end{pmatrix} = \begin{pmatrix} 0.8148 \\ 0.5828 \end{pmatrix} - \begin{pmatrix} 0.0105 \\ -0.0083 \end{pmatrix} = \begin{pmatrix} 0.8043 \\ 0.5911 \end{pmatrix}.$$

Поскольку далее $\bar{f}(\bar{x}^{(5)}) = \begin{pmatrix} -0.0001 \\ -0.0013 \end{pmatrix}$, вычислительный процесс можно закончить, утверждая, что заданная точность вычисления корней исходной системы достигнута.

6.7. Лабораторная работа № 11. Решение систем нелинейных уравнений методом Ньютона

Основная идея метода Ньютона состоит в выделении из уравнений системы

$\bar{f}(\bar{x}) = 0$ линейных частей, которые являются главными при малых приращениях аргументов. Это позволяет свести одну исходную задачу к решению последовательности задач для линейных систем.

Итерационная формула метода Ньютона для системы нелинейных уравнений (6.4.7) имеет вид $\bar{x}^{(k+1)} = \bar{x}^{(k)} - \Phi^{-1}\left(\bar{x}^{(k)}\right) \cdot \bar{f}\left(\bar{x}^{(k)}\right)$. Необходимость обращения матрицы первых частных производных при каждой итерации сильно затрудняет решение. Эти затруднения чаще носят технический характер, тем не менее вместо уравнения (6.4.7) иногда решают систему линейных алгебраических уравнений вида (6.4.8):

$$\begin{cases} \Phi\left(\bar{x}^{(k)}\right) \cdot \Delta\bar{x}^{(k+1)} = \bar{f}\left(\bar{x}^{(k)}\right) \\ \bar{x}^{(k+1)} = \bar{x}^{(k)} + \Delta\bar{x}^{(k+1)}. \end{cases}$$

По методу Ньютона итерационный процесс при наличии хорошего начального приближения сходится с квадратичной скоростью, то есть если $\|\bar{x}^{(k+1)} - \bar{x}^{(k)}\| < \varepsilon$, то $\|\bar{x}^{(k+1)} - \bar{x}_{\text{точн.}}\| \approx \varepsilon^2 \ll \varepsilon$.

В пакете Mathcad для решения систем нелинейных уравнений служат конструкции **Given-Find** и **Given-MinErr**, разобранные в четвертой лабораторной работе. С помощью этих подпрограмм можно решать системы объемом до двухсот уравнений.

Рассмотрим **пример**. Пусть дана система

$$\begin{cases} \tan(xy + 0.3) - x^2 = 0, \\ \frac{x^2}{2} + 2y^2 = 1. \end{cases}$$

Вводим программу

$$\text{ORIGIN} := 1 \quad f1(x, y) := \tan(xy + 0.3) - x^2 \quad f2(x, y) := 0.5 \cdot x^2 + 2 \cdot y^2 - 1$$

$$x_0 := \begin{pmatrix} -0.5 \\ 1.0 \end{pmatrix} \quad A(x, y) := \begin{pmatrix} \frac{d}{dx} f1(x, y) & \frac{d}{dy} f1(x, y) \\ \frac{d}{dx} f2(x, y) & \frac{d}{dy} f2(x, y) \end{pmatrix} \quad f(x, y) := \begin{pmatrix} f1(x, y) \\ f2(x, y) \end{pmatrix}$$

$$A(x_{0_1}, x_{0_2}) = \begin{pmatrix} 2.0410914 & -0.5205457 \\ -0.5 & 4 \end{pmatrix} \quad f(x_{0_1}, x_{0_2}) = \begin{pmatrix} -0.4527100 \\ 1.125 \end{pmatrix}$$

$$x := -0.5 \quad y := 1.0$$

$$\text{Given } \tan(xy + 0.3) - x^2 = 0$$

$$0.5 \cdot x^2 + 2 \cdot y^2 - 1 = 0$$

$$\text{Find}(x, y) = \begin{pmatrix} -0.3023399 \\ 0.6907587 \end{pmatrix}$$

$$\text{Given } \tan(xy + 0.3) - x^2 = 0$$

$$0.5 \cdot x^2 + 2 \cdot y^2 - 1 = 0$$

$$\text{MinErr}(x, y) = \begin{pmatrix} -0.3023399 \\ 0.6907587 \end{pmatrix}$$

При задании иных начальных условий получаются другие точки экстремума. Например,

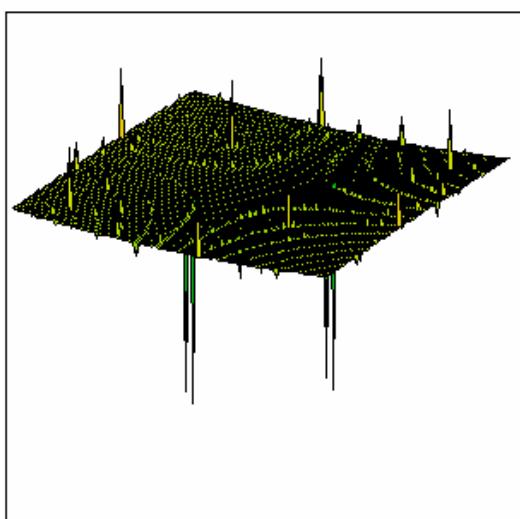
$$x := 1.0 \quad y := 1.0$$

$$\begin{aligned} \text{Given } & \tan(x \cdot y + 0.3) - x^2 = 0 \\ & 0.5 \cdot x^2 + 2 \cdot y^2 - 1 = 0 \\ \text{Find } & (x, y) = \begin{pmatrix} 1.0144947 \\ 0.4926461 \end{pmatrix} \end{aligned}$$

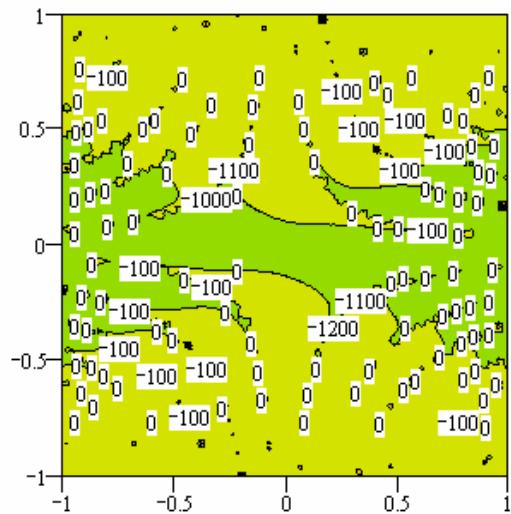
$$x := -1.0 \quad y := 3.0$$

$$\begin{aligned} \text{Given } & \tan(x \cdot y + 0.3) - x^2 = 0 \\ & 0.5 \cdot x^2 + 2 \cdot y^2 - 1 = 0 \\ \text{MinErr } & (x, y) = \begin{pmatrix} -2.1256772 \\ 0.9774855 \end{pmatrix} \end{aligned}$$

Это неудивительно, ибо поверхность, представляющая функцию, равную сумме уравнений исходной системы, имеет множество локальных экстремумов. Убедимся в этом, построив эту поверхность и ее линии уровня:



M3



M3

$$\begin{aligned} i &:= 1 \dots 100 \quad j := 1 \dots 100 \quad x_i := -5 + 0.1 \cdot i \quad y_j := -5 + 0.1 \cdot j \\ M3_{i,j} &:= f1(x_i, y_j) + f2(x_i, y_j) \end{aligned}$$

Найти решение упрощенным методом Ньютона по формулам (6.6.1), когда матрица $\Phi^{-1}\left(\frac{\partial}{\partial x^{(k)}}\right)$ не перевычисляется на каждом шаге, можно с помощью нескольких операторов.

$$\begin{aligned} \Phi &:= A(x_0)_1, A(x_0)_2 \quad \Phi^{-1} = \begin{pmatrix} 0.506 & 0.066 \\ 0.063 & 0.258 \end{pmatrix} \\ xv^{(1)} &:= x_0 \quad i := 1 \dots 11 \quad xv^{(i+1)} := xv^{(i)} - \Phi^{-1} \cdot f\left(\left(xv^{(i)}\right)_1, \left(xv^{(i)}\right)_2\right) \end{aligned}$$

	8	9	10	11	12
xv =	-0.3024526	-0.3023817	-0.3023553	-0.3023456	-0.302342
	0.690798	0.6907712	0.6907627	0.69076	0.6907592

Наконец, подпрограммы, реализующие вычисления по методу Ньютона по формулам (6.4.7) и (6.4.8)-(6.4.9), могут быть такими:

```

Newtsyst(f,A,n,x0,t) := | norm-> $\sqrt{|x_0|}$ 
                           | for i $\in$  1..n
                           |   x1i->x0i
                           |   fun->f(x11,x12)
                           |   while norm>t
                           |     B->A(x11,x12)
                           |     Φ->B-1
                           |     for i $\in$  1..n
                           |       x2i->x1i -  $\sum_{j=1}^n \Phi_{i,j} \cdot \text{fun}_j$ 
                           |       fun->f(x21,x22)
                           |       norm-> $\sqrt{|fun|}$ 
                           |   x1->x2
                           | x2

```

```

Nlinter(f,A,n,x0,t) := | norm-> $\sqrt{|x_0|}$ 
                           | for i $\in$  1..n
                           |   x1i->x0i
                           |   while norm>t
                           |     fun->-f(x11,x12)
                           |     B->A(x11,x12)
                           |     x2->lsolve(B,fun)
                           |     norm-> $\sqrt{|x2|}$ 
                           |     x2->x1 + x2
                           |   x1->x2
                           | x2

```

Параметры этих подпрограмм одинаковы:

$$f = \bar{f}(x) = \begin{pmatrix} f_1(x_1, x_2, \dots, x_n) \\ f_2(x_1, x_2, \dots, x_n) \\ \dots \\ f_n(x_1, x_2, \dots, x_n) \end{pmatrix},$$

$A = \begin{pmatrix} \frac{\partial f_1}{\partial x_1} & \frac{\partial f_1}{\partial x_2} & \dots & \frac{\partial f_1}{\partial x_n} \\ \frac{\partial f_2}{\partial x_1} & \frac{\partial f_2}{\partial x_2} & \dots & \frac{\partial f_2}{\partial x_n} \\ \dots & \dots & \dots & \dots \\ \frac{\partial f_n}{\partial x_1} & \frac{\partial f_n}{\partial x_2} & \dots & \frac{\partial f_n}{\partial x_n} \end{pmatrix}$, n - размерность системы, $x0$ - вектор начального приближения,

ϵ - заданная точность вычислений. Приведенные варианты подпрограмм рассчитаны на систему из двух уравнений с двумя неизвестными, однако легко переделываются для любого ко-

нечного числа неизвестных и уравнений. Для этого нужно лишь перечислить требуемое число аргументов в операторах, вычисляющих вектор $\bar{f}(\bar{x})$ и матрицу A внутри подпрограмм.

Вводим конец программы:

$$\varepsilon := 10^{-6} \quad xx := \text{Newtsyst}(f, A, 2, x0, \varepsilon) \quad xx = \begin{pmatrix} -0.3023399 \\ 0.6907587 \end{pmatrix}$$

$$xy := \text{Nlinter}(f, A, 2, x0, \varepsilon) \quad xy = \begin{pmatrix} -0.3023399 \\ 0.6907587 \end{pmatrix}$$

$$x0 := \begin{pmatrix} 1.0 \\ 1.0 \end{pmatrix} \quad xx := \text{Newtsyst}(f, A, 2, x0, \varepsilon) \quad xx = \begin{pmatrix} -0.3023399 \\ 0.6907587 \end{pmatrix}$$

$$x0 := \begin{pmatrix} 2.0 \\ 2.0 \end{pmatrix} \quad xx := \text{Newtsyst}(f, A, 2, x0, \varepsilon) \quad xx = \begin{pmatrix} -0.3023399 \\ 0.6907587 \end{pmatrix}$$

$$x0 := \begin{pmatrix} 5.0 \\ 2.0 \end{pmatrix} \quad xx := \text{Newtsyst}(f, A, 2, x0, \varepsilon) \quad - \text{ подпрограмма не может решить систему, начальное приближение плохое и процесс расходится.}$$

Задание № 1. Любым способом, разобранным в этой лабораторной работе, решить методом Ньютона следующую систему уравнений:

$$1. \begin{cases} \operatorname{tg}(xy) = x^2, \\ 0.8x^2 + 2y^2 = 1. \end{cases}$$

$$2. \begin{cases} \cos(y-1) + x = 0.8, \\ y - \cos x = 2. \end{cases}$$

$$3. \begin{cases} \sin(x+y) - 1.3x = 0, \\ x^2 + y^2 = 1. \end{cases}$$

$$4. \begin{cases} \sin(x+0.5) - y = 1, \\ \cos(y-2) + x = 0. \end{cases}$$

$$5. \begin{cases} \operatorname{tg}(xy + 0.4) = x^2, \\ 0.6x^2 + 2y^2 = 1, \quad x > 0, \quad y > 0. \end{cases}$$

$$6. \begin{cases} \cos(y+0.5) + x = 0.8, \\ \sin x - 2y = 1.6. \end{cases}$$

$$7. \begin{cases} \sin(y-1) + x = 1.3, \\ y - \sin(x+1) = 0.8. \end{cases}$$

$$8. \begin{cases} \sin(x-y) - xy = -1, \\ x^2 - y^2 = \frac{3}{4}. \end{cases}$$

$$9. \begin{cases} \sin(y+1) - x = 1, \\ 2y + \cos x = 2. \end{cases}$$

$$10. \begin{cases} \operatorname{tg}(xy) = x^2, \\ 0.5x^2 + 2y^2 = 1. \end{cases}$$

$$11. \begin{cases} \sin(x+y) = 1.5x + 0.2, \\ x^2 + y^2 = 1. \end{cases}$$

$$12. \begin{cases} \sin(x+y) - 1.6x = 0, \\ x^2 + y^2 = 1, \quad x > 0, \quad y > 0. \end{cases}$$

$$13. \begin{cases} 2y - \cos(x+1) = 0, \\ x + \sin y = -0.4. \end{cases}$$

$$14. \begin{cases} \sin(y+2) - y = 1.5, \\ x + \cos(y-2) = 0.5. \end{cases}$$

$$15. \begin{cases} \operatorname{tg}(xy + 0.1) = x^2, \\ x^2 + 2y^2 = 1. \end{cases}$$

$$16. \begin{cases} \cos(x-1) + y = 0.8, \\ x - \cos y = 2. \end{cases}$$

$$17. \begin{cases} \sin(x+y) - 1.1x = 0.1, \\ x^2 + y^2 = 1. \end{cases}$$

$$18. \begin{cases} \sin(x-0.6) - y = 1.6, \\ 3x - \cos y = 0.9. \end{cases}$$

$$19. \begin{cases} \operatorname{tg}(xy + 0.1) = x^2, \\ 0.9x^2 + 2y^2 = 1. \end{cases}$$

$$20. \begin{cases} \sin(2x-y) - 1.2x = 0.4, \\ 0.8x^2 + 1.5y^2 = 1. \end{cases}$$

21. $\begin{cases} \operatorname{tg}(xy) = x^2, \\ 0.7x^2 + 2y^2 = 1. \end{cases}$
23. $\begin{cases} \cos y + x = 1.5, \\ 2y - \sin(x - 0.5) = 1. \end{cases}$
25. $\begin{cases} \operatorname{tg}(xy + 0.2) = x^2, \\ x^2 + y^2 = 1. \end{cases}$
27. $\begin{cases} \cos(x - 1) + y = 1, \\ \sin y + 2x = 1.6. \end{cases}$
29. $\begin{cases} \cos x + y = 1.2, \\ 2x - \sin(y - 0.5) = 2. \end{cases}$

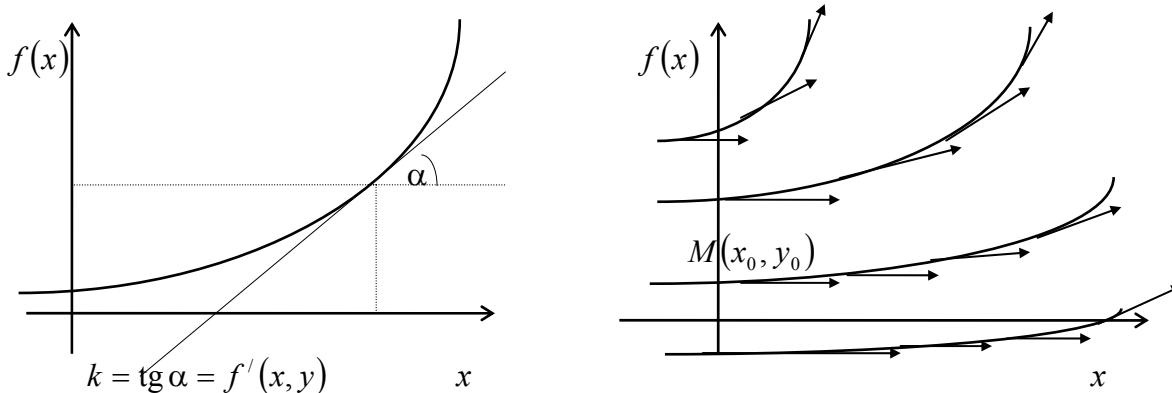
22. $\begin{cases} \cos(x + 0.5) + y = 1, \\ \sin y - 2x = 2. \end{cases}$
24. $\begin{cases} \operatorname{tg}(x - y) - xy = 0, \\ x^2 + 2y^2 = 1. \end{cases}$
26. $\begin{cases} \sin(x - 1) + y = 1.5, \\ x - \sin(y + 1) = 1. \end{cases}$
28. $\begin{cases} 2x - \cos(y + 1) = 0, \\ y - \sin x = -0.4. \end{cases}$
30. $\begin{cases} \operatorname{tg}(xy) = x^2, \\ 0.6x^2 + 2y^2 = 1. \end{cases}$

7. ЧИСЛЕННЫЕ МЕТОДЫ РЕШЕНИЯ ЗАДАЧИ КОШИ ДЛЯ ОБЫКНОВЕННЫХ ДИФФЕРЕНЦИАЛЬНЫХ УРАВНЕНИЙ И СИСТЕМ ДИФФЕРЕНЦИАЛЬНЫХ УРАВНЕНИЙ

7.1. Задача Коши для дифференциального уравнения первого порядка

Большая часть инженерных задач приводит к законам и правилам, которые формулируются в виде дифференциальных уравнений. Основная задача, решаемая для таких уравнений, - это задача Коши или начальная задача. В подавляющем большинстве случаев она решается с использованием вычислительной техники. Рассмотрим методы решения дифференциальных уравнений первого порядка.

Решением обыкновенного дифференциального уравнения первого порядка $y'(x) = f(x, y(x))$ называется функция $y(x)$, обращающая это уравнение в тождество при подстановке. График решения называется интегральной кривой.



На рисунке справа изображено поле направлений, представляющее семейство интегральных кривых уравнения

$$y'(x) = f(x, y(x)). \quad (7.1.1)$$

Задание начального условия - точки $M(x_0, y_0)$ выделяет из этого семейства конкретную кривую, дающую частное решение уравнения (7.1.1).

Задача нахождения при $x > x_0$ решения $y(x)$ дифференциального уравнения (7.1.1), удовлетворяющего начальному условию

$$y(x_0) = y_0, \quad (7.1.2)$$

называется задачей Коши. Чаще всего решение ищут на конечном заданном отрезке $[x_0, b]$.

Теорема 7.1. (Теорема существования и единственности решения). Пусть функция $f(x, y)$ определена и непрерывна в области G и удовлетворяет в ней условию Липшица* по переменной y , то есть

$$|f(x, y_1) - f(x, y_2)| \leq c|y_1 - y_2|, \quad c = \text{const}. \quad (7.1.3)$$

Тогда для любого начального значения y_0 существует единственное решение $y(x)$ задачи Коши с заданным начальным условием

$$\begin{cases} y' = f(x, y(x)), \\ y(x_0) = y_0, \end{cases}$$

определенное на отрезке $[x_0, b] \subset G$.

На практике для дифференцируемых по y функций $f(x, y)$ условие (7.1.3) заменяется более грубым условием

$$|f'_y(x, y)| \leq c, \quad (7.1.4)$$

то есть условием ограниченности частной производной.

7.2. Численные методы решения задачи Коши. Основные понятия и определения

Численное решение задачи Коши строится для ее дискретного аналога. В этом случае отрезок $[x_0, X]$ - область непрерывности изменения аргумента x заменяется множеством $\bar{\omega}^h$ - конечным множеством точек $x_0 < x_1 < \dots < x_N = X$, которое называется сеткой. Величина $h_n = x_n - x_{n-1}$ - шаг сетки - является, как правило, постоянным, то есть сетки в большинстве случаев равномерные.

Функции, определенные лишь в узлах сетки $\bar{\omega}^h$, называются сеточными. Они помечаются индексом h , например, f^h , чаще же значение функции $f(x)$ в узлах сетки $\bar{\omega}^h$ обозначается обычным образом с помощью индекса, например, $f(x_n)$ или f_n .

В основе построения конкретного численного метода лежит тот или иной способ замены дифференциального уравнения $y' = f(x, y)$ его дискретным аналогом - уравнением вида

$$\frac{1}{h} \sum_{j=0}^k a_j y_{n+1-j} = \Phi(x_n, y_{n+1-k}, \dots, y_n, y_{n+1}, h), \quad (7.2.1)$$

где y^h - значения сеточной функции в $k+1$ последовательных точках $x_{n+1-k}, \dots, x_n, x_{n+1}$. Сумма в левой части формулы (7.2.1) рассматривается как разностная аппроксимация производной y' по одной из формул численного дифференцирования, а правая часть - как специальным образом построенная аппроксимация функции $f(x, y)$.

При нахождении приближения y_{n+1} в очередной точке сетки по формуле (7.2.1) используются найденные ранее значения сеточной функции y^h в k предыдущих точках x_{n+1-k}, \dots, x_n . Такие методы называются k -шаговыми. При $k = 1$ уравнение (7.2.1) принимает вид

$$\frac{y_{n+1} - y_n}{h} = \Phi(x_n, y_n, y_{n+1}, h). \quad (7.2.2)$$

Соответствующий этой формуле метод называется одношаговым. Вычисление y_{n+1} осуществляется здесь с использованием только одного предыдущего значения y_n .

* Рудольф Липшиц (1832 - 1903) - немецкий математик.

В случае, когда входящая в уравнение (7.2.1) функция Φ не зависит от y_{n+1} , вычисление y_{n+1} не вызывает затруднений и осуществляется по явной формуле

$$y_{n+1} = a_0^{-1} \left[- \sum_{j=1}^k a_j y_{n+1-j} + h\Phi(x_n, y_{n+1-k}, \dots, y_n, h) \right]. \quad (7.2.3)$$

Соответствующие методы называются явными. Напротив, если Φ зависит от y_{n+1} , на каждом шаге приходится решать относительно y_{n+1} нелинейное уравнение (7.2.1). Методы, реализующие такой алгоритм, называются неявными.

7.3. Решение с помощью рядов Тейлора

Начнем с метода, который теоретически пригоден для решения любых дифференциальных уравнений, но с вычислительной точки зрения не представляет почти никакого практического интереса. Его ценность заключается в том, что он дает некоторый эталон для сравнения различных практически удобных методов.

Запишем разложение функции $y(x)$ в ряд Тейлора в окрестности точки x_{n+1} :

$$y(x_{n+1}) = y(x_n + h) = y_n + \frac{y'_n}{1!}(x_{n+1} - x_n) + \frac{y''_n}{2!}(x_{n+1} - x_n)^2 + \frac{y'''_n}{3!}(x_{n+1} - x_n)^3 + \dots, \quad (7.3.1)$$

где $y_n^{(j)}$ есть j -я производная функции $y(x)$ в точке x_n . Пусть имеется приближенное решение уравнения (7.3.1) для $n+1$ точки $x_0, x_1, \dots, x_n, x_i = x_0 + ih, h = \text{const}$. Найдем приближенное решение для точки x_{n+1} , подставив $x_{n+1} = x_n + h$ в формулу (7.3.1). Получим

$$y_{n+1} = y_n + hy'_n + \frac{h^2}{2}y''_n + \frac{h^3}{6}y'''_n + \dots \quad (7.3.2)$$

Чем больше членов ряда (7.3.2) взять для вычислений, тем точнее будет приближение. В любом случае необходимо вычислять различные производные функции $y(x)$. Из (7.1.1) имеем $y'_n = f(x_n, y_n)$. Дифференцируя по x , получим

$$y'' = \frac{\partial}{\partial x} f(x, y) + \frac{\partial}{\partial y} f(x, y) \cdot y' \quad \text{или} \quad y''_n = f'_x + f \cdot f'_y. \quad (7.3.3)$$

При этом уравнение (7.3.2) приобретает вид

$$y_{n+1} = y_n + h \left(f + \frac{h}{2} (f'_x + f \cdot f'_y) \right) + O(h^3) \quad (7.3.4)$$

Процедура нахождения решения с помощью ряда Тейлора является одношаговым методом, так как для вычисления y_{n+1} требуется информация только об одной предыдущей точке (x_n, y_n) .

На практике очень трудно иногда найти f'_x и f'_y . Кроме того, если попытаться получить лучшее приближение, то есть меньшую ошибку, то необходимо вычислить y'''_n , которая равна:

$$y'''_n = f''_{xx} + 2f \cdot f''_{xy} + f^2 \cdot f''_{yy} + f'_x \cdot f'_y + f \cdot f''_{xy}. \quad (7.3.5)$$

Последующие производные становятся еще более сложными, а их вычисление чаще всего является весьма трудоемкой и даже невыполнимой операцией.

7.4. Методы Рунге - Кутты*

Методы Рунге - Кутты обладают следующими отличительными свойствами:

* Мартин Вильгельм Кутта (1867-1944) - немецкий физик и математик.

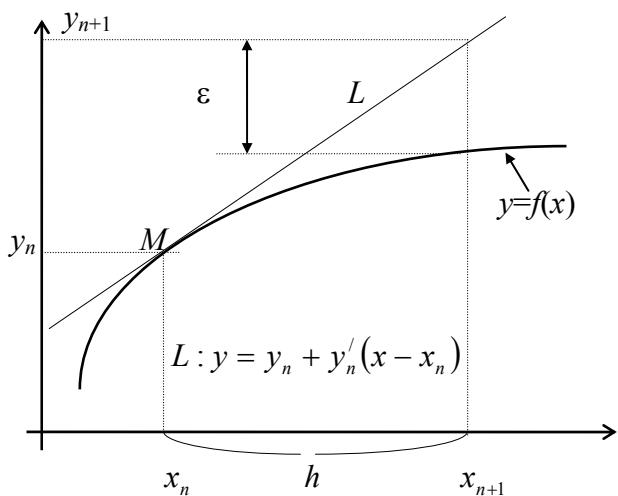
1) являются одношаговыми: чтобы найти y_{n+1} нужна информация только о предыдущей точке (x_n, y_n) ;

2) согласуются с рядом Тейлора вплоть до членов порядка h^p , где степень p различна для различных методов и называется порядком метода;

3) не требуют вычисления производных от $f(x, y)$, а только вычисления функции.

Именно благодаря третьему свойству методы Рунге - Кутты более известны, нежели ряд Тейлора. Однако для вычисления одной последующей точки решения приходится вычислять $f(x, y)$ несколько раз при различных значениях x и y .

Выведем сначала некоторые формулы на основе геометрических аналогий.



вания дифференциальных уравнений. Формула (7.4.1) может быть получена из (7.2.2), если принять $\Phi(x_n, y_n, y_{n+1}, h) \equiv f(x_n, y_n)$. Так как здесь функция Φ не зависит от y_{n+1} , то метод является явным.

Ошибка интегрирования при $x = x_{n+1}$ показана на рисунке в виде отрезка ε . Очевидно, что найденное таким образом приближенное решение согласуется с разложением в ряд Тейлора вплоть до членов порядка h , так что ошибка равна $\varepsilon = Kh^2$.

Теорема 7.2. Пусть функция $f(x, y)$ удовлетворяет условию $|f_y'| \leq M$. Тогда справедливо неравенство

$$\max_{0 \leq n \leq N} |y_n - y_n^*| \leq \exp(M(X - x_0)) \cdot \left(|y_0 - y_0^*| + h \sum_{k=0}^{N-1} |\psi_k| \right), \quad (7.4.2)$$

то есть метод Эйлера устойчив на конечном отрезке. Здесь $\psi_n = \frac{1}{h} \sum_{j=0}^k a_j y(x_{n+1-j}) - \Phi(x_n, y(x_{n+1-k}), \dots, y(x_{n+1}), h)$ - погрешность аппроксимации дискретного уравнения (7.2.1) на решении $y(x)$.

Метод Эйлера, реализуемый формулой (7.4.1), можно усовершенствовать множеством различных способов. Рассмотрим две модификации: а) исправленный метод Эйлера и б) модифицированный метод Эйлера.

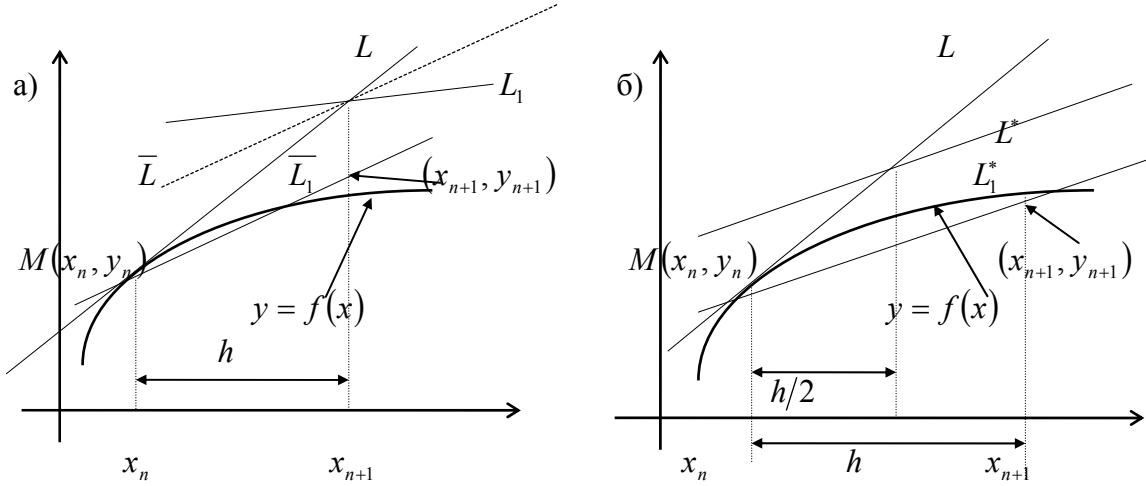
а). В исправленном методе Эйлера находим средний тангенс угла наклона касательной для двух точек (x_n, y_n) и $(x_n + h, y_n + hy'_n)$. Геометрический процесс нахождения точки (x_{n+1}, y_{n+1}) можно проследить по левому рисунку на следующей странице. С помощью метода Эйлера находится точка $(x_n + h, y_n + hy'_n)$, лежащая на прямой L . В этой точке снова вы-

Пусть известна точка $M(x_n, y_n)$ на искомой кривой. Через эту точку можно провести прямую с тангенсом угла наклона $y'_n = f(x_n, y_n)$. Тогда следующей можно считать точку, где прямая L пересечет ординату, проведенную через точку $x = x_{n+1} = x_n + h$. Уравнение прямой L имеет вид $y = y_n + y'_n(x - x_n)$, но так как $y'_n = f(x_n, y_n)$ и $x_{n+1} = x_n + h$, то $y_{n+1} = y_n + hf(x_n, y_n)$. (7.4.1)

Формула (7.4.1) описывает метод Эйлера*, один из самых старых и широко известных методов численного интегриро-

* Леонард Эйлер (1707-1783) - швейцарский математик. Долгое время жил и работал в России.

числяется тангенс угла наклона касательной, на рисунке этому значению соответствует прямая L_1 . Усреднение двух тангенсов дает прямую \bar{L} . Наконец, через точку (x_n, y_n) проводим прямую \bar{L}_1 , параллельную \bar{L} . Точка, в которой прямая \bar{L}_1 пересечется с ординатой x_{n+1} и будет искомой точкой (x_{n+1}, y_{n+1}) .



Тангенс угла наклона прямой \bar{L} равен:

$$\Phi(x_n, y_n, h) = \frac{1}{2} [f(x_n, y_n) + f(x_n + h, y_n + hy'_n)], \quad y'_n = f(x_n, y_n). \quad (7.4.3)$$

Уравнение \bar{L}_1 при этом записывается в виде $y = y_n + (x - x_n)\Phi(x_n, y_n, h)$. Таким образом,

$$y_{n+1} = y_n + h\Phi(x_n, y_n, h) = y_n + \frac{h}{2} [f(x_n, y_n) + f(x_n + h, y_n + hy'_n)]. \quad (7.4.4)$$

Это и есть рабочее уравнение исправленного метода Эйлера.

Выясним, как хорошо этот метод согласуется с разложением в ряд Тейлора. Для этого запишем разложение в ряд Тейлора для функции двух переменных в окрестности точки $M(x_n, y_n)$: $f(x, y) = f(x_n, y_n) + (x - x_n) \frac{\partial f(x_n, y_n)}{\partial x} + (y - y_n) \frac{\partial f(x_n, y_n)}{\partial y} + \dots$. Если положить здесь $x = x_n + h$ и $y = y_n + hy'_n$, $y'_n = f(x_n, y_n)$, то получим

$$f(x_n + h, y_n + hy'_n) = f + hf'_x + hf \cdot f'_y + O(h^2).$$

Подставляя этот результат в (7.4.3) и производя необходимые преобразования, будем иметь $\Phi(x_n, y_n, h) = f + \frac{h}{2} (f'_x + f \cdot f'_y) + O(h^2)$, что совпадает с (7.3.2) вплоть до членов степени h^2 .

Таким образом, исправленный метод Эйлера является методом Рунге - Кутты второго порядка.

б). Если в рассмотренном методе усреднялись наклоны касательных, то в модифицированном методе Эйлера усредняются точки (смотрите рисунок справа). Первоначальное построение сделано точно так же, как и в предыдущем случае - через точку $M(x_n, y_n)$ проведена прямая L с тангенсом угла наклона, равным $f(x_n, y_n)$. Затем взята точка на пересечении этой прямой и ординаты $x = x_n + \frac{h}{2}$. Угол наклона касательной L^* в этой точке

$$\Phi(x_n, y_n, h) = f\left(x_n + \frac{h}{2}, y_n + \frac{h}{2} y'_n\right). \quad (7.4.5)$$

Проведем через точку $M(x_n, y_n)$ прямую L_1^* , параллельную L^* . Пересечение этой прямой с ординатой $x = x_n + h$ и даст исковую точку (x_{n+1}, y_{n+1}) . Так как уравнение прямой L_1^* можно записать в виде $y = y_n + (x - x_n)\Phi(x_n, y_n, h)$, то

$$y_{n+1} = y_n + h\Phi(x_n, y_n, h) = y_n + hf\left(x_n + \frac{h}{2}, y_n + \frac{h}{2}y'_n\right). \quad (7.4.6)$$

Формула (7.4.6) описывает модифицированный метод Эйлера.

В литературе исправленный метод Эйлера называют иногда методом Эйлера - Коши*, а модифицированный метод - усовершенствованным. Как и в предыдущем случае, можно легко показать, что модифицированный метод является методом Рунге - Кутты второго порядка.

Оба рассмотренных метода описываются формулами вида $y_{n+1} = y_n + h\Phi(x_n, y_n, h)$, причем в обоих случаях функция Φ имеет вид

$$\Phi(x_n, y_n, h) = a_1 f(x_n, y_n) + a_2 f(x_n + b_1 h, y_n + b_2 h y'_n), \quad y'_n = f(x_n, y_n). \quad (7.4.7)$$

Для исправленного метода Эйлера $\begin{cases} a_1 = a_2 = \frac{1}{2}, \\ b_1 = b_2 = 1, \end{cases}$ а для модифицированного $\begin{cases} a_1 = 0, a_2 = 1, \\ b_1 = b_2 = \frac{1}{2}. \end{cases}$

Методы Рунге - Кутты третьего и четвертого порядков можно вывести совершенно аналогично тому, как это делалось при выводе методов первого и второго порядков. Общая идея вывода формулы метода Рунге - Кутты любого заданного порядка состоит в следующем.

Пусть $y(x)$ - решение дифференциального уравнения $y'(x) = f(x, y(x))$, удовлетворяющее условию $y(x_n) = y_n$. Проинтегрируем уравнение $y' = f(x, y)$ по x от x_n до x_{n+1} , полу

чим $\int_{x_n}^{x_{n+1}} y'(x) dx = \int_{x_n}^{x_{n+1}} f(x, y(x)) dx$. По формуле Ньютона - Лейбница**

$$\int_{x_n}^{x_{n+1}} y'(x) dx = y(x_{n+1}) - y(x_n). \quad \text{Тогда } y_{n+1} = y_n + \int_{x_n}^{x_{n+1}} f(x, y(x)) dx. \quad (7.4.8)$$

Если бы интеграл в формуле (7.4.8) вычислялся точно, то она была бы основной рабочей формулой всех методов численного интегрирования дифференциальных уравнений. В действительности используют приближенную формулу, заменяя интеграл квадратурной суммой.

Введем на отрезке $[x_n, x_{n+1}]$ m вспомогательных узлов $x_n^{(1)} = x_n + a_1 h$, $x_n^{(2)} = x_n + a_2 h$, ..., $x_n^{(m)} = x_n + a_m h$, где $0 = a_1 \leq a_2 \leq \dots \leq a_m \leq 1$. Тогда интеграл в уравнении (7.4.8) можно заменить квадратурной суммой с узлами $x_n^{(1)}, x_n^{(2)}, \dots, x_n^{(m)}$, то есть

$$y(x_{n+1}) \approx y(x_n) + h \sum_{i=1}^m c_i f(x_n^{(i)}, y(x_n^{(i)})). \quad (7.4.9)$$

Здесь неизвестны значения $y(x_n^{(i)})$. Применяя формулу (7.4.8), получим

$$y(x_n^{(i)}) = y(x_n) + \int_{x_n}^{x_n^{(i)}} f(x, y(x)) dx, \quad i = 2, 3, \dots, m.$$

* Огюст Луи Коши (1789-1857.) - французский математик.

** Готфрид Вильгельм Лейбница (1646 - 1716.) - немецкий математик, создатель дифференциального исчисления.

Заменим для каждого i входящий в эту формулу интеграл соответствующей ему квадратурной суммой с узлами $x_n^{(1)}, x_n^{(2)}, \dots, x_n^{(m)}$:

$$\left\{ \begin{array}{l} y(x_n^{(2)}) \approx y(x_n) + hb_{21}f(x_n^{(1)}, y(x_n^{(1)})), \\ y(x_n^{(3)}) \approx y(x_n) + h(b_{31}f(x_n^{(1)}, y(x_n^{(1)})) + b_{32}f(x_n^{(2)}, y(x_n^{(2)}))), \\ \dots \\ y(x_n^{(i)}) \approx y(x_n) + h \sum_{j=1}^{i-1} b_{ij}f(x_n^{(j)}, y(x_n^{(j)})), \\ \dots \\ y(x_n^{(m)}) \approx y(x_n) + h \sum_{j=1}^{m-1} b_{mj}f(x_n^{(j)}, y(x_n^{(j)})). \end{array} \right. \quad (7.4.10)$$

Формулы (7.4.10) позволяют последовательно вычислять приближения к значениям $y(x_n^{(2)}), y(x_n^{(3)}), \dots, y(x_n^{(m)})$. Пусть $y_n^{(i)} = y(x_n^{(i)})$, $k_n^{(i)} = f(x_n^{(i)}, y_n^{(i)})$. Тогда формулу (7.4.8) можно переписать в виде

$$\left\{ \begin{array}{l} y_{n+1} = y_n + hk_n, \quad k_n = \sum_{i=1}^m c_i k_n^{(i)}, \\ k_n^{(1)} = f(x_n^{(1)}, y_n^{(1)}), \quad y_n^{(1)} = y_n, \\ k_n^{(2)} = f(x_n^{(2)}, y_n^{(2)}), \quad y_n^{(2)} = y_n + hb_{21}k_n^{(1)}, \\ \dots \\ k_n^{(m)} = f(x_n^{(m)}, y_n^{(m)}), \quad y_n^{(m)} = y_n + h \sum_{j=1}^{m-1} b_{mj}k_n^{(j)}. \end{array} \right. \quad (7.4.11)$$

Если исключить отсюда величины $y_n^{(i)}$, получим

$$\left\{ \begin{array}{l} y_{n+1} = y_n + hk_n, \quad k_n = \sum_{i=1}^m c_i k_n^{(i)}, \\ k_n^{(1)} = f(x_n, y_n), \\ k_n^{(2)} = f\left(x_n + a_2 h, y_n + hb_{21}k_n^{(1)}\right), \\ \dots \\ k_n^{(m)} = f\left(x_n + a_m h, y_n + h \sum_{j=1}^{m-1} b_{mj}k_n^{(j)}\right). \end{array} \right. \quad (7.4.12)$$

Выбор конкретных значений параметров c_i, a_i и b_i осуществляется по-разному и дает ту или иную модификацию методов Рунге - Кутты. Приведем рабочие формулы метода четвертого порядка. Он применяется настолько широко, что в литературе называется просто «методом Рунге - Кутты» без всяких указаний на тип или порядок. Этот классический метод Рунге - Кутты описывается системой следующих шести уравнений:

$$\left\{ \begin{array}{l} y_{n+1} = y_n + hk_n, \quad k_n = \frac{1}{6} (k_n^{(1)} + 2k_n^{(2)} + 2k_n^{(3)} + k_n^{(4)}), \\ k_n^{(1)} = f(x_n, y_n), \\ k_n^{(2)} = f\left(x_n + \frac{h}{2}, y_n + \frac{h}{2}k_n^{(1)}\right), \\ k_n^{(3)} = f\left(x_n + \frac{h}{2}, y_n + \frac{h}{2}k_n^{(2)}\right), \\ k_n^{(4)} = f(x_n + h, y_n + hk_n^{(3)}). \end{array} \right. \quad (7.4.13)$$

Ошибка метода $\varepsilon = Kh^5$, при его использовании функцию необходимо вычислять дважды.

7.5. Анализ ошибок, возникающих при использовании методов Рунге - Кутты

Один из серьезных недостатков методов Рунге - Кутты состоит в отсутствии простых способов оценки ошибки интегрирования. Все же без некоторой оценки ошибки трудно правильно выбрать величину шага интегрирования h . Применим так называемый принцип Рунге.

Пусть $y_n^{(T)}$ есть точное решение дифференциального уравнения $y' = f(x, y)$ при $x = x_0 + nh$. Тогда для метода Рунге - Кутты, описываемого формулами (7.4.13), справедлива следующая оценка погрешности:

$$\varepsilon^{\left(\frac{h}{2}\right)} = \frac{y_n^{\left(\frac{h}{2}\right)} - y_n^{(h)}}{2^p - 1}, \quad (7.5.1)$$

где $y_n^{(h)}$ - приближение к точному решению $y_n^{(T)}$, вычисленное с шагом h , $y_n^{\left(\frac{h}{2}\right)}$ - такое же приближение с шагом $\frac{h}{2}$, $\varepsilon^{\left(\frac{h}{2}\right)} = y_n^{(T)} - y_n^{\left(\frac{h}{2}\right)}$.

Так как для метода, описываемого формулами (7.4.13), $p = 4$, то

$$\varepsilon^{\left(\frac{h}{2}\right)} = \frac{1}{15} \left[y_n^{\left(\frac{h}{2}\right)} - y_n^{(h)} \right]. \quad (7.5.2)$$

Формула (7.5.1) выведена в предположении, что на каждом шаге интегрирования допускается погрешность, приблизительно пропорциональная h^{p+1} , то есть $y_n^{(T)} = y_n^{(h)} + Kh^5$, что справедливо для достаточно гладких функций. Таким образом, ошибка интегрирования в предположении, что $y^{(5)}(x)$ практически постоянна, равна:

$$\varepsilon^{\left(\frac{h}{2}\right)} = Kh^5 = \frac{1}{15} \left[y_n^{\left(\frac{h}{2}\right)} - y_n^{(h)} \right].$$

Это довольно точная оценка, однако для ее использования необходимо вычислять решение дважды.

Предложено несколько полуэмпирических критериев смены шага и выбора оптимального шага интегрирования при условии достижения заданной точности. Например, используя

ется такое оценочное правило: если $\frac{|k_n^{(2)} - k_n^{(3)}|}{|k_n^{(1)} - k_n^{(2)}|}$ достаточно велико (обычно больше нескольких сотых), то шаг интегрирования необходимо уменьшить.

Существуют более точные методы оценки погрешности интегрирования, основанные на использовании для контроля точности двух различных методов Рунге - Кутты. Один из самых эффективных - метод Рунге - Кутты - Фельберга. В этом методе для оценки погрешности метода пятого порядка используются формулы метода четвертого порядка точности, причем на одном шаге интегрирования требуется всего лишь шесть вычислений значений правой части $f(x, y)$.

7.6. Методы прогноза и коррекции

Отличительной чертой методов Рунге - Кутты является то, что при вычислении следующей точки (x_{n+1}, y_{n+1}) используется информация только о предыдущей (x_n, y_n) . Зато приходится вычислять значение правой части в нескольких промежуточных точках. Это нерационально, поскольку при наложенном процессе интегрирования имеется дополнительная информация - предыдущие точки решения.

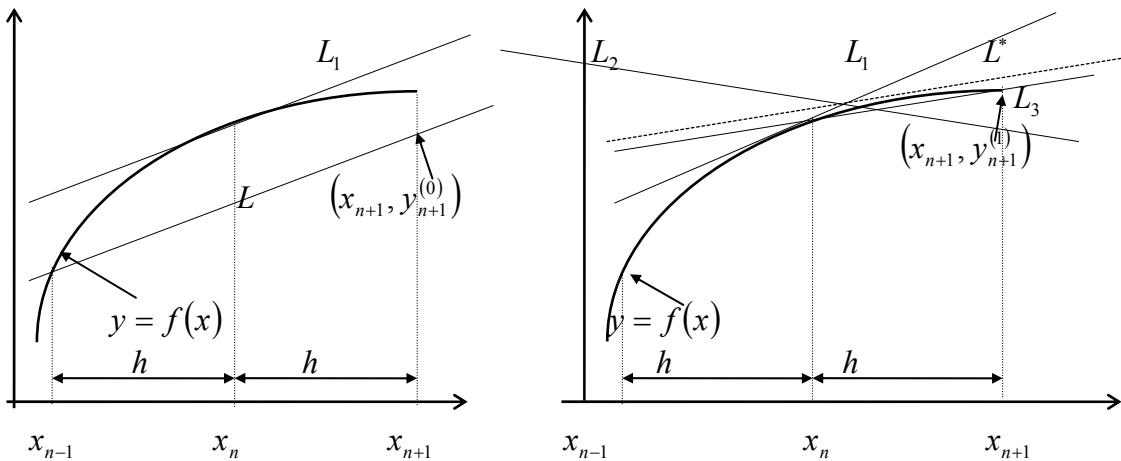
Рассмотрим общую идею группы методов, известных в литературе под названием «методов прогноза и коррекции». Как ясно из названия, вначале «предсказывается» значение y_{n+1} , а затем используется тот или иной метод для «корректировки» этого значения. Этую же формулу можно использовать сколько угодно раз для повторной корректировки уже скорректированного значения y_{n+1} .

Для демонстрации основных идей метода можно использовать для прогноза формулу любого метода численного интегрирования. Воспользуемся формулой второго порядка

$$y_{n+1}^{(0)} = y_{n-1} + 2hf(x_n, y_n), \quad (7.6.1)$$

где индекс (0) означает исходное приближение к y_{n+1} . Непосредственно из написанной формулы следует, что с ее помощью нельзя вычислить, например, y_1 , ибо для этого потребовалась бы точка, расположенная перед начальной точкой y_0 . Чтобы начать решение по методу прогноза и коррекции, часто используется метод Рунге - Кутты.

Геометрически предсказание сводится к тому, что находится угол наклона касательной в точке (x_n, y_n) (см. левый рисунок). После этого через точку (x_{n-1}, y_{n-1}) проводится



прямая L , параллельная L_1 . Предсказанное значение $y_{n+1}^{(0)}$ будет расположено там, где прямая L пересечется с абсциссой $x = x_{n+1}$. Скорректируем теперь предсказанное значение. Вычислим наклон касательной в точке $(x_{n+1}, y_{n+1}^{(0)})$. Эта касательная изображена на правом рисунке и обозначена L_2 . Усредним тангенсы углов наклона прямых L_1 и L_2 и получим прямую L^* . Наконец, проведем через точку (x_n, y_n) прямую L_3 , параллельную L^* , и точка пересечения этой прямой с абсциссой $x = x_{n+1}$ даст новое приближение $(x_{n+1}, y_{n+1}^{(1)})$.

Это значение вычисляется по формуле

$$y_{n+1}^{(1)} = y_n + \frac{h}{2} [f(x_n, y_n) + f(x_{n+1}, y_{n+1}^{(0)})] \quad (7.6.2)$$

В общем случае i -е приближение, очевидно, будет находиться таким образом:

$$y_{n+1}^{(i)} = y_n + \frac{h}{2} [f(x_n, y_n) + f(x_{n+1}, y_{n+1}^{(i-1)})] \quad (7.6.3)$$

Итерационный процесс можно прекратить, когда

$$|y_{n+1}^{(i+1)} - y_{n+1}^{(i)}| < \varepsilon. \quad (7.6.4)$$

К настоящему времени разработано много методов прогноза и коррекции разных порядков, обеспечивающих очень большую точность решения. В некоторых из них по заданной точности предусмотрен автоматический выбор шага и автоматический выбор порядка метода.

7.7. Сравнение методов

а). Методы Рунге - Кутты.

1. В этих методах используется информация только об очередной точке решения, поэтому с их помощью можно начинать решение.

2. По той же самой причине приходится многократно вычислять функцию $f(x, y)$.

3. Методы этой группы позволяют очень легко менять величину шага h .

4. При использовании этих методов трудно получить оценку ошибки интегрирования.

б). Методы прогноза и коррекции.

1. Так как в этих методах используется информация о ранее вычисленных точках решения, то с их помощью нельзя начать решение.

2. Из-за использования информации о ранее вычисленных точках методы этой группы более экономичны по затратам машинного времени.

3. При любом изменении величины шага h приходится временно возвращаться к методам Рунге - Кутты.

4. В качестве побочного продукта вычислений получается хорошая оценка ошибки интегрирования.

Как всегда, наиболее целесообразным является использование при решении практических задач комбинации этих двух методов.

Пример. Методом Рунге - Кутты с шагом $h = 0.1$ найти на отрезке $[0, 0.3]$ решение следующего дифференциального уравнения: $y' = \frac{\cos bx}{a + y^2}$ с начальным условием $y(0) = 0$, $a = 1.4$, $b = 2.6$.

Обозначим через y_i приближенное значение решения в точке x_i . Формулы метода:

$$\left\{ \begin{array}{l} y_{i+1} = y_i + hk_i, \quad k_i = \frac{1}{6} \left(k_i^{(1)} + 2k_i^{(2)} + 2k_i^{(3)} + k_i^{(4)} \right), \\ k_i^{(1)} = f(x_i, y_i), \\ k_i^{(2)} = f\left(x_i + \frac{h}{2}, y_i + \frac{h}{2}k_i^{(1)}\right), \\ k_i^{(3)} = f\left(x_i + \frac{h}{2}, y_i + \frac{h}{2}k_i^{(2)}\right), \\ k_i^{(4)} = f\left(x_i + h, y_i + hk_i^{(3)}\right) \end{array} \right. \quad (7.7.1)$$

Все вычисления удобно располагать по схеме, указанной в таблице.

i	x	y	k	Δy
0	x_0	y_0	$k_0^{(1)}$	$k_0^{(1)}$
	$x_0 + \frac{h}{2}$	$y_0 + \frac{h}{2}k_0^{(1)}$	$k_0^{(2)}$	$2k_0^{(2)}$
	$x_0 + \frac{h}{2}$	$y_0 + \frac{h}{2}k_0^{(2)}$	$k_0^{(3)}$	$2k_0^{(3)}$
	$x_0 + h$	$y_0 + hk_0^{(3)}$	$k_0^{(4)}$	$k_0^{(4)}$
				Δy_0
1	x_1	y_1		

Она заполняется следующим образом.

1. Записываем в первой строке x_0 , y_0 , вычисляем $f(x_0, y_0)$ и заносим в таблицу в качестве $k_0^{(1)}$.

2. Для второй строки вычисляем $x_0 + \frac{h}{2}$ и $y_0 + \frac{h}{2}k_0^{(1)}$, затем находим $f\left(x_0 + \frac{h}{2}, y_0 + \frac{h}{2}k_0^{(1)}\right)$ и записываем в качестве $k_0^{(2)}$.

3. Находим $y_0 + \frac{h}{2}k_0^{(2)}$, вычисляем $f\left(x_0 + \frac{h}{2}, y_0 + \frac{h}{2}k_0^{(2)}\right)$ и записываем в таблицу на место $k_0^{(3)}$.

4. Записываем в четвертой строке $x_0 + h$ и $y_0 + hk_0^{(3)}$, затем находим $k_0^{(4)} = f(x_0 + h, y_0 + hk_0^{(3)})$

5. Суммируем числа, стоящие в столбце Δy , делим на шесть и заносим в таблицу в качестве Δy_0 .

6. Вычисляем $x_1 = x_0 + h$, $y_1 = y_0 + \Delta y_0$. Затем все вычисления необходимо повторить с пункта 1, принимая за начальную точку (x_1, y_1) . Для контроля правильности выбора шага h рекомендуется вычислять дробь $\theta = \frac{|k_i^{(2)} - k_i^{(3)}|}{|k_i^{(1)} - k_i^{(2)}|}$. Величина θ не должна превышать двух

- трех сотых. В противном случае шаг h следует уменьшить.

i	x	y	k	Δy	θ
0	0.00	0.000000	0.714286	0.14286	
	0.05	0.035714	0.707614	1.415228	
	0.05	0.035381	0.707626	1.415252	
	0.10	0.070763	0.687818	0.687818	0.0018
				0.705431	
1	0.10	0.705431	0.509261	0.509262	
	0.15	0.730894	0.478185	0.956370	
	0.15	0.729340	0.478747	0.957494	
	0.20	0.753306	0.441084	0.441084	0.0018
				0.477368	
2	0.20	1.182799	0.310045	0.310045	
	0.25	1.198301	0.280714	0.561428	
	0.25	1.196835	0.281062	0.562124	
	0.30	1.210905	0.248026	0.248026	0.0012
				0.280270	
3	0.30	1.463069			

7.8. Лабораторная работа № 12. Методы интегрирования обыкновенных дифференциальных уравнений

Существует большое число методов приближенного решения дифференциальных уравнений, основанных на самых различных идеях. Численные методы дают приближенное решение $y(x)$ в виде таблицы значений y_1, y_2, \dots, y_n в точках x_1, x_2, \dots, x_n .

Простейшим методом численного интегрирования дифференциального уравнения первого порядка $y' = f(x, y)$, удовлетворяющего начальному условию $y(x_0) = y_0$, является метод Эйлера. В нем величины y_i вычисляются по формуле

$$\begin{cases} x_i = x_0 + ih, & i = 0, 1, 2, \dots \\ y_{i+1} = y_i + hf(x_i, y_i). \end{cases}$$

Метод Эйлера относится к группе одношаговых, в которых для расчета точки (x_{i+1}, y_{i+1}) требуется информация только о последней вычисленной точке (x_i, y_i) . Геометрическая интерпретация метода изложена в подразд. 7.4.

В среде Mathcad имеется тринацать встроенных функций решения дифференциальных уравнений и систем (задача Коши, краевая задача, уравнения в частных производных). Самая употребительная из них - **rkfixed**, в которую заложен метод Рунге – Кутты четвертого порядка с постоянным шагом. Подпрограммы для метода Эйлера нет из-за его низкой точности. Формулы метода Эйлера настолько просты, что вычисления по ним можно организовать с помощью дискретной переменной.

Рассмотрим **пример**. Решим задачу Коши для дифференциального уравнения $y' = \cos(x - y) + \frac{1.25y}{1.5 + x}$ при $y(0) = 0$.

Введем начало программы

```
ORIGIN := 1  f(x,y) := cos(x - y) + 1.25 ·  $\frac{y}{1.5 + x}$   a := 0  b := 1  h := 0.1  y0 := 0
i := 1...11  xi := a + h · (i - 1)
i := 2...11  yi := yi-1 + h · f(xi-1, yi-1)
```

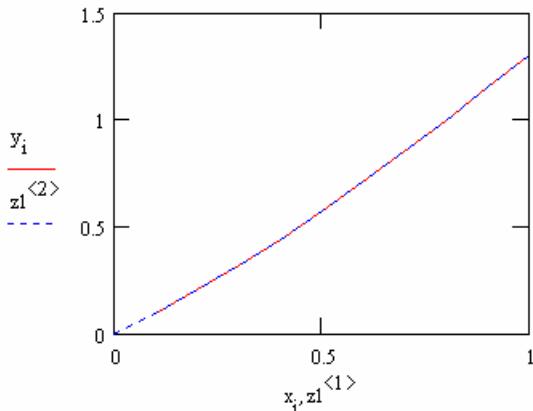
x^T	0.0	0.1	0.2	0.3	0.4	0.5	0.6	0.7	0.8	0.9	1.0
y^T	0.000	0.100	0.208	0.323	0.445	0.575	0.710	0.852	0.999	1.152	1.308

Аналогичного результата можно достигнуть, введя подпрограмму, реализующую формулы (7.4.1) метода Эйлера:

```
Eiler(y0, a, b, h, f) := 
$$\begin{cases} n \leftarrow \frac{(b - a)}{h} \\ z_{1,1} \leftarrow a \\ z_{1,2} \leftarrow y0 \\ \text{for } i \in 2..n+1 \\ \quad \begin{cases} z_{i,1} \leftarrow z_{i-1,1} + h \\ z_{i,2} \leftarrow z_{i-1,2} + h \cdot f(z_{i-1,1}, z_{i-1,2}) \end{cases} \\ z \end{cases}$$

```

$z1 := Eiler(y0, 0, 1, 0.1, f)$

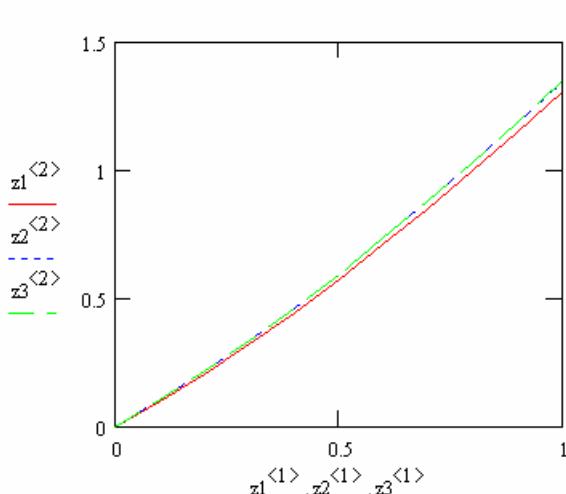


	1	2
1	0	0
2	0.1	0.1
3	0.2	0.208
4	0.3	0.323
5	0.4	0.445
6	0.5	0.575
7	0.6	0.71
8	0.7	0.852
9	0.8	0.999
10	0.9	1.152
11	1	1.308

Минимальная переделка подпрограммы позволяет запрограммировать исправленный и модифицированный метод Эйлера по формулам (7.4.4) и (7.4.6).

$$\begin{array}{ll}
 \text{Ieiler}(y_0, a, b, h, f) := & \left| \begin{array}{l} n \leftarrow \frac{(b - a)}{h} \\ z_{1,1} \leftarrow a \\ z_{1,2} \leftarrow y_0 \\ \text{for } i \in 2..n+1 \\ \quad \left| \begin{array}{l} z_{i,1} \leftarrow z_{i-1,1} + h \\ c \leftarrow f(z_{i-1,1}, z_{i-1,2}) \\ d \leftarrow f(z_{i,1}, z_{i-1,2} + h \cdot c) \\ z_{i,2} \leftarrow z_{i-1,2} + \frac{h}{2} \cdot (c + d) \end{array} \right. \\ z \end{array} \right. \\
 & \text{Meiler}(y_0, a, b, h, f) := \left| \begin{array}{l} n \leftarrow \frac{(b - a)}{h} \\ z_{1,1} \leftarrow a \\ z_{1,2} \leftarrow y_0 \\ \text{for } i \in 2..n+1 \\ \quad \left| \begin{array}{l} z_{i,1} \leftarrow z_{i-1,1} + h \\ c \leftarrow f(z_{i-1,1}, z_{i-1,2}) \\ d \leftarrow f\left(z_{i-1,1} + \frac{h}{2}, z_{i-1,2} + \frac{h}{2} \cdot c\right) \\ z_{i,2} \leftarrow z_{i-1,2} + h \cdot d \end{array} \right. \\ z \end{array} \right.
 \end{array}$$

$$z2 := \text{Ieiler}(y0, 0, 1, 0.1, f) \quad z3 := \text{Meiler}(y0, 0, 1, 0.1, f)$$

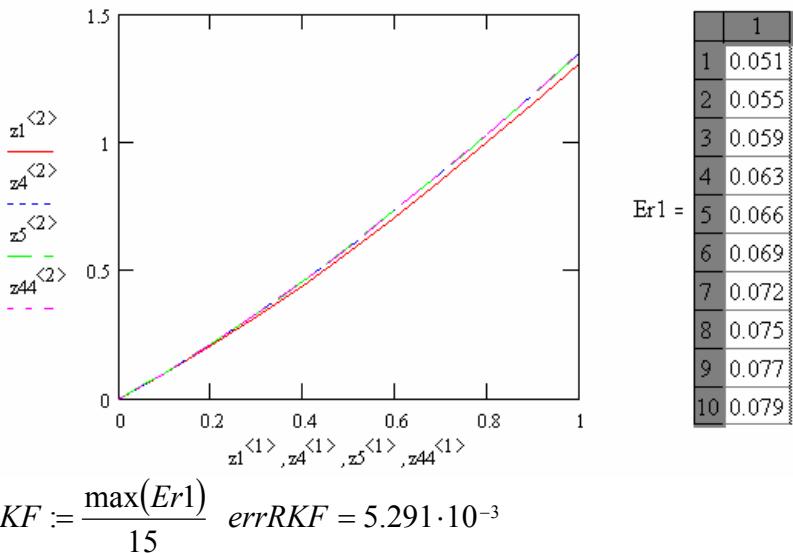


	1	2		1	2	
1	0	0		1	0	0
2	0.1	0.104		2	0.1	0.104
3	0.2	0.216		3	0.2	0.216
4	0.3	0.335		4	0.3	0.336
5	0.4	0.462		5	0.4	0.462
6	0.5	0.595		6	0.5	0.596
7	0.6	0.735		7	0.6	0.736
8	0.7	0.88		8	0.7	0.881
9	0.8	1.031		9	0.8	1.032
10	0.9	1.187		10	0.9	1.188
11	1	1.346		11	1	1.348

Обратимся теперь к средствам пакета Mathcad. Для решения обыкновенных «неособенных» дифференциальных уравнений здесь используются две функции **rkfixed** и **Rkadapt**, реализующие метод Рунге – Кутты четвертого порядка с постоянным и переменным шагом. Набор параметров у этих подпрограмм одинаков: $\text{rkfixed}(y, a, b, n, f)$, где $y = y(x_0)$, a и b - начало и конец интервала интегрирования, n - число точек и, следовательно, шаг $h = \frac{b-a}{n}$, $f(x, y)$ - правая часть дифференциального уравнения. Несмотря на то что при решении дифференциального уравнения функция **Rkadapt** использует переменный шаг, она тем не менее представляет ответ для n точек, находящихся на одинаковом расстоянии друг от друга, равном $h = \frac{b-a}{n}$.

Вводим следующую часть программы:

$$\begin{aligned}
 yy_1 &:= 0 \quad \text{fun}(x, yy) := \cos(x - yy_1) + 1.25 \cdot \frac{yy_1}{1.5 + x} \\
 z4 &:= \text{rkfixed}(yy, 0, 1, 10, \text{fun}) \quad z5 := \text{rkfixed}(yy, 0, 1, 20, \text{fun}) \\
 z4 &:= \text{Rkadapt}(yy, 0, 1, 10, \text{fun}) \\
 i &:= 1..10 \quad Er1_i := |(z4^{(2)})_i - (z5^{(2)})_{2:i}|
 \end{aligned}$$



Обращение к **rkfixed** с $h_1 = 0.1$ и $h_2 = 0.05$ сделано для оценки погрешности интегрирования по правилу Рунге (7.5.2). $errRKF$ - оценка погрешности.

На данном учебном примере, конечно, невозможно оценить выгоды использования функции **Rkadapt** вместо **rkfixed**. В более сложных случаях **Rkadapt** решает уравнение более точно и быстро. Даже в этом примере точность решения по **Rkadapt** выше, чем по **rkfixed**. Это видно из следующей таблицы:

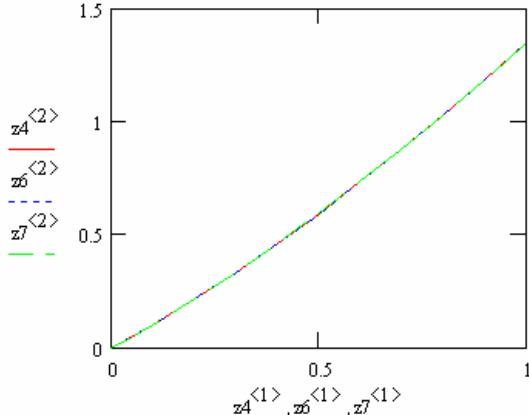
x	0.1	0.2	0.3	0.4	0.5
z_4	0.1040989	0.2161356	0.3357322	0.4625076	0.5960572
z_5	0.1040990	0.2161359	0.3357326	0.4625081	0.5960578
z_{44}	0.1040990	0.2161359	0.3357326	0.4625081	0.5960571
x	0.6	0.7	0.8	0.9	1.0
z_4	0.7359363	0.8816484	1.0326377	1.1882891	1.3479326
z_5	0.7359370	0.8816491	1.0326386	1.1882900	1.3479335
z_{44}	0.7359371	0.8816492	1.0326386	1.1882901	1.3479336

В заключение приведем подпрограмму **RGK**, реализующую формулы (7.4.13) метода Рунге – Кутты четвертого порядка. Далее, как и в предыдущем случае, произведена оценка погрешности по правилу Рунге и даны графики полученных решений. Программа метода Рунге – Кутты четвертого порядка (**RGK**) приведена после графиков проинтегрированных функций.

$$z6 := RGK(y0, 0, 1, 0.1, f) \quad z7 := RGK(y0, 0, 1, 0.05, f)$$

$$i := 1 \dots 10 \quad Er2_i := \left| (z6^{(2)})_i - (z7^{(2)})_{2-i} \right|$$

$$errRGK := \frac{\max(Er2)}{15} \quad errRGK = 5.291 \cdot 10^{-3}$$



```

RGK(y0, a, b, h, f) := | n ← (b - a)
                         |   h
                         | z1,1 ← a
                         | z1,2 ← y0
                         | for i ∈ 2..n + 1
                         |   k1 ← f(zi-1,1, zi-1,2)
                         |   c ← zi-1,1 +  $\frac{h}{2}$ 
                         |   d ← zi-1,2 +  $\frac{h}{2} \cdot k1$ 
                         |   k2 ← f(c, d)
                         |   d ← zi-1,2 +  $\frac{h}{2} \cdot k2$ 
                         |   k3 ← f(c, d)
                         |   zi,1 ← zi-1,1 + h
                         |   k4 ← f(zi,1, zi-1,2 + h · k3)
                         |   c ←  $\frac{1}{6} \cdot (k1 + 2 \cdot (k2 + k3) + k4)$ 
                         |   zi,2 ← zi-1,2 + h · c
                         |
                         | z

```

Задание № 1. Решить заданное дифференциальное уравнение первого порядка методом Эйлера и Рунге – Кутты четвертого порядка на отрезке $x \in [0, 1]$ с шагом $h = 0.1$ и оценить погрешность интегрирования по правилу Рунге.

1. $y' = 1 + 0.2y \sin x - y^2$, $y(0) = 0.1$.
2. $y' = \cos(x + y) + 0.5(x - y)$, $y(0) = 0$.
3. $y' = \frac{\cos x}{x + 1} - 0.5y^2$, $y(0) = 0.2$.
4. $y' = (1 - y^2)\cos x + 0.6y$, $y(0) = 0$.
5. $y' = 1 + 0.4y \sin x - 1.5y^2$, $y(0) = 1$.
6. $y' = \frac{\cos y}{x + 2} + 0.3y^2$, $y(0) = 0$.
7. $y' = \cos(1.5x + y) + (x - y)$, $y(0) = 0.5$.
8. $y' = 1 - \sin(x + y) + \frac{0.5y}{x + 2}$, $y(0) = 0.3$.
9. $y' = \frac{\cos y}{1.5 + x} + 0.1y^2$, $y(0) = 0$.
10. $y' = 0.6 \sin x - 1.25y^2 + 1$, $y(0) = 1$.
11. $y' = \cos(2x + y) + 1.5(x - y)$, $y(0) = 0.1$.
12. $y' = 1 - \frac{0.1y}{x + 2} - \sin(2x + y)$, $y(0) = 0.5$.
13. $y' = \frac{\cos y}{1.25 + x} - 0.1y^2$, $y(0) = 0.2$.
14. $y' = 1 + 0.8y \sin x - 2y^2$, $y(0) = 0.5$.

$$15. y' = \cos(1.5x + y) + 1.5(x - y), y(0) = 1.$$

$$16. y' = 1 - \sin(2x + y) + \frac{0.3y}{x+2}, y(0) = 0.$$

$$17. y' = \frac{\cos y}{1.75+x} - 0.5y^2, y(0) = 0.1.$$

$$18. y' = 1 + (1-x)\sin y - (2+x)y, y(0) = 0.5.$$

$$19. y' = (0.8 - y^2)\cos x + 0.3y, y(0) = 0.$$

$$20. y' = 1 + 2.2\sin x + 1.5y, y(0) = 0.$$

$$21. y' = \cos(x + y) + 0.75(x - y), y(0) = 0.3.$$

$$22. y' = 1 - \sin(1.25x + y) + \frac{0.5y}{x+2}, y(0) = 0.5.$$

$$23. y' = \frac{\cos y}{x+2} - 0.3y^2, y(0) = 0.$$

$$24. y' = 1 - \sin(1.75x + y) + \frac{0.1y}{x+2}, y(0) = 1.$$

$$25. y' = \frac{\cos y}{1.25+x} - 0.5y^2, y(0) = 0.$$

$$26. y' = \cos(1.5x + y) - 2.25(x + y), y(0) = 2.$$

$$27. y' = \frac{\cos y}{1.5+x} - 1.25y^2, y(0) = 1.$$

$$28. y' = 1 - (x - 1)\sin y + 2(x + y), y(0) = 0.5.$$

$$29. y' = 1 - \sin(0.75x - y) + \frac{1.75y}{x+1}, y(0) = 0.$$

$$30. y' = \cos(x - y) + \frac{1.25y}{1.5+x}, y(0) = 0.5.$$

7.9. Решение задачи Коши для систем обыкновенных дифференциальных уравнений

Как правило, на практике приходиться решать задачу Коши не для одного дифференциального уравнения, а для системы вида

$$\begin{cases} y'_1(x) = f_1(x, y_1(x), y_2(x), \dots, y_n(x)), \\ y'_2(x) = f_2(x, y_1(x), y_2(x), \dots, y_n(x)), \\ \dots \\ y'_n(x) = f_n(x, y_1(x), y_2(x), \dots, y_n(x)). \end{cases} \quad (7.9.1)$$

Здесь $y_1(x), y_2(x), \dots, y_n(x)$ - искомые функции, значения которых подлежат определению на $[x_0, X]$. В момент $x = x_0$ задаются начальные условия

$$\begin{cases} y_1(x_0) = y_{10}, \\ y_2(x_0) = y_{20}, \\ \dots \\ y_n(x_0) = y_{n0}, \end{cases} \quad (7.9.2)$$

определяющие начальное состояние системы (7.9.1).

Введем вектор-функции $\bar{y}(x) = (y_1(x), y_2(x), \dots, y_n(x))^T$, $\bar{f}(x, \bar{y}) = (f_1(x, \bar{y}), f_2(x, \bar{y}), \dots, f_n(x, \bar{y}))^T$ и вектор $\bar{y}_0 = (y_{10}, y_{20}, \dots, y_{n0})^T$. Тогда задачу Коши (7.9.1), (7.9.2) можно записать в компактной форме:

$$\begin{cases} \bar{y}'(x) = \bar{f}(x, \bar{y}), \\ \bar{y}(x_0) = \bar{y}_0. \end{cases} \quad (7.9.3)$$

Теорема 7.3. (Теорема существования и единственности решения). Пусть вектор - функция $\bar{f}(x, \bar{y})$ определена и непрерывна в области G_x и удовлетворяет условию Липшица $\|\bar{f}(x, \bar{y}_1) - \bar{f}(x, \bar{y}_2)\| \leq L \|\bar{y}_1 - \bar{y}_2\|$, $L = \text{const}$, $L > 0$, $(7.9.4)$

для $\forall x \in [x_0, X]$ и произвольных \bar{y}_1, \bar{y}_2 . Тогда для каждого начального значения \bar{y}_0 существует единственное решение $\bar{y}(x)$ задачи Коши, определенное на отрезке $[x_0, X]$.

Если функции f_1, f_2, \dots, f_n непрерывно дифференцируемы по y_1, y_2, \dots, y_n , то условие (7.9.4) эквивалентно условию $\|f'_y(x, \bar{y})\| \leq L$, где матрица Якоби

$$f'_y(x, \bar{y}) = \begin{pmatrix} f'_{1,y_1}(x, \bar{y}) & f'_{1,y_2}(x, \bar{y}) & \dots & f'_{1,y_n}(x, \bar{y}) \\ f'_{2,y_1}(x, \bar{y}) & f'_{2,y_2}(x, \bar{y}) & \dots & f'_{2,y_n}(x, \bar{y}) \\ \dots & \dots & \dots & \dots \\ f'_{n,y_1}(x, \bar{y}) & f'_{n,y_2}(x, \bar{y}) & \dots & f'_{n,y_n}(x, \bar{y}) \end{pmatrix}. \quad (7.9.5)$$

На практике методы решения задачи Коши одного уравнения можно использовать и для систем, причем уравнения претерпевают минимальные изменения. Следует лишь заменить в расчетных формулах числа y_i на векторы $\bar{y}_i = (y_1, y_2, \dots, y_n)^T$, а функцию f - на вектор - функцию \bar{f} .

Например, расчетная формула метода Эйлера $y_{i+1} = y_i + hf(x_i, y_i)$ применительно к системе (7.9.3) примет вид

$$\begin{cases} \bar{y}_{i+1} = \bar{y}_i + h\bar{f}(x_i, \bar{y}_i) \text{ или} \\ y_{1,i+1} = y_{1i} + hf_1(x_i, y_{1i}, y_{2i}, \dots, y_{ni}), \\ y_{2,i+1} = y_{2i} + hf_2(x_i, y_{1i}, y_{2i}, \dots, y_{ni}), \\ \dots \\ y_{n,i+1} = y_{ni} + hf_n(x_i, y_{1i}, y_{2i}, \dots, y_{ni}). \end{cases} \quad (7.9.6)$$

Метод Рунге - Кутты четвертого порядка точности (7.4.13) порождает следующие формулы:

$$\begin{cases} \bar{y}_{i+1} = \bar{y}_i + h\bar{k}_i, \quad \bar{k}_i = \frac{1}{6}(\bar{k}_i^{(1)} + 2\bar{k}_i^{(2)} + 2\bar{k}_i^{(3)} + \bar{k}_i^{(4)}) \\ \bar{k}_i^{(1)} = \bar{f}(x_i, \bar{y}_i), \quad \bar{k}_i^{(2)} = \bar{f}\left(x_i + \frac{h}{2}, \bar{y}_i + \frac{h}{2}\bar{k}_i^{(1)}\right), \\ \bar{k}_i^{(3)} = \bar{f}\left(x_i + \frac{h}{2}, \bar{y}_i + \frac{h}{2}\bar{k}_i^{(2)}\right), \quad \bar{k}_i^{(4)} = \bar{f}\left(x_i + h, \bar{y}_i + h\bar{k}_i^{(3)}\right) \end{cases} \quad (7.9.7)$$

Теория численных методов решения задачи Коши для систем дифференциальных уравнений имеет много общего с соответствующей теорией решения задачи Коши для одного дифференциального уравнения.

7.10. Сведение задачи Коши для уравнения n -го порядка к задаче Коши для системы уравнений первого порядка

Рассмотрим дифференциальное уравнение n -го порядка в нормальной форме

$$y^{(n)}(x) = f(x, y(x), y'(x), \dots, y^{(n-1)}(x))$$

с начальными условиями

$$\begin{cases} y(x_0) = y_{10}, \\ y'(x_0) = y_{20}, \\ \dots \\ y^{(n-1)}(x_0) = y_{n0}. \end{cases} \quad (7.10.1)$$

Положим $y_1(x) = y(x)$, $y_2(x) = y'(x), \dots, y_n(x) = y^{(n-1)}(x)$. Кроме того, очевидно, что $y_k(x) = y'_{k-1}(x)$, например, $y_2(x) = y'(x) = y'_1(x)$, $y_3(x) = y''(x) = (y'(x))' = (y'_1(x))' = y''_2(x)$ и так далее. Тогда уравнение (7.10.1) с заданными начальными условиями примет вид

$$\begin{cases} y'_1(x) = y_2(x), \\ y'_2(x) = y_3(x), \\ \dots \\ y'_{n-1}(x) = y_n(x), \\ y'_n(x) = f(x, y_1(x), y_2(x), \dots, y_n(x)). \end{cases} \quad (7.10.2)$$

Начальные условия в новых обозначениях будут выглядеть так:

$$\begin{cases} y_1(x_0) = y_{10}, \\ y_2(x_0) = y_{20}, \\ \dots \\ y_n(x_0) = y_{n0}. \end{cases} \quad (7.10.3)$$

Чаще всего при необходимости решить уравнение (7.10.1) его приводят к виду (7.10.2) и (7.10.3), а далее решают систему дифференциальных уравнений. Редко, но бывают случаи, когда сведение к системе (7.10.2) не требуется. Обычно иными способами решают дифференциальные уравнения специального вида.

Пример. Методом Рунге–Кутты решить дифференциальное уравнение $y'' - xy' + 2xy = 0.8$ с начальными условиями $\begin{cases} y(1.5) = 1, \\ y'(1.5) = 2, \end{cases}$ на отрезке $x \in [1.5, 1.8]$ с шагом $h = 0.1$.

Сведем это уравнение к системе дифференциальных уравнений первого порядка. Пусть $y(x) = y_1(x)$, $y'(x) = y_2(x)$. Тогда $y'(x) = y'_1(x) = y_2(x)$, $y''(x) = (y'(x))' = y'_2(x) = 0.8 - 2xy_1 + xy_2$. Таким образом, нужная система имеет вид

$$\begin{cases} y'_1(x) = y_2(x), \\ y'_2(x) = 0.8 - 2xy_1(x) + xy_2(x), \end{cases} \quad (7.10.4)$$

при начальных условиях $\begin{cases} y_1(1.5) = 1, \\ y_2(1.5) = 2. \end{cases} \quad (7.10.5)$

Удобнее всего все вычисления для $y_1(x)$ и $y_2(x)$ помещать в одну таблицу. Опишем подробно последовательность действий при заполнении этой таблицы при $i = 0$.

1. При $i = 0$ записываем $x_0 = 1.5$, а в столбец $(y_1(x_i), y_2(x_i))$ заносим

$y_1(x_0) = 1$, $y_2(x_0) = 2$. Вычисляем $\begin{cases} k_0^{(1)}_{y_1} = f_1(x_0, y_{10}, y_{20}) = y_{20} = 2, \\ k_0^{(1)}_{y_2} = f_2(x_0, y_{10}, y_{20}) = 0.8 - 2x_0y_{10} + x_0y_{20} = 0.8. \end{cases}$ Эти величины $k_0^{(1)}$ заносим также в столбец Δf , так как в формуле интегрирования они используются с множителем единица.

2. Для $x = x_0 + \frac{h}{2} = 1.55$ в столбце $(y_1(x_i), y_2(x_i))$ записываем

$$\begin{cases} y_{10} + \frac{h}{2} k_0^{(1)}_{y_1} = 1 + 0.05 \cdot 2 = 1.100000, \\ y_{20} + \frac{h}{2} k_0^{(1)}_{y_2} = 2 + 0.05 \cdot 0.8 = 2.040000. \end{cases}$$

Вычисляем для следующего столбца и записываем в него $k_0^{(2)}_{y_1} = 2.040000$,

$$k_0^{(2)}_{y_2} = 0.8 - 2 \cdot \left(x_0 + \frac{h}{2} \right) \cdot \left(y_{10} + \frac{h}{2} k_0^{(1)}_{y_1} \right) + \left(x_0 + \frac{h}{2} \right) \cdot \left(y_{20} + \frac{h}{2} k_0^{(1)}_{y_2} \right) = 0.8 - 2 \cdot 1.55 \cdot 1.10 + 1.55 \cdot 2.04 = 0.552000. В столбец \Delta f записываем k_0^{(2)}_{y_1} и k_0^{(2)}_{y_2} с множителем два (см. формулу (7.4.13)).$$

3. Следующие две строки таблицы заполняются аналогично пункту два. Именно, в столбец x записываем $x_0 + \frac{h}{2} = 1.55$. Для столбца $(y_1(x_i), y_2(x_i))$ находим

$$\begin{cases} y_{10} + \frac{h}{2} k_0^{(2)}_{y_1} = 1 + 0.05 \cdot 2.040000 = 1.102000, \\ y_{20} + \frac{h}{2} k_0^{(2)}_{y_2} = 2 + 0.05 \cdot 0.552000 = 2.027600. \end{cases}$$

Для следующего столбца вычисляем величины $k_0^{(3)}_{y_1} = y_{20} + \frac{h}{2} k_0^{(2)}_{y_2} = y_2 = 2.027600$, $k_0^{(3)}_{y_2} = 0.8 - 2 \cdot 1.55 \cdot 1.102 + 1.55 \cdot 2.0276 = 0.526580$. На конец, в столбец Δf ставим числа $2k_0^{(3)}_{y_1}$ и $2k_0^{(3)}_{y_2}$.

4. В столбец записываем $x_0 + h = 1.60$. В столбец $(y_1(x_i), y_2(x_i))$ записываем

$$\begin{cases} y_{10} + hk_0^{(3)}_{y_1} = 1 + 0.1 \cdot 2.027600 = 1.202760, \\ y_{20} + hk_0^{(3)}_{y_2} = 2 + 0.1 \cdot 0.526580 = 2.052658. \end{cases}$$

Далее вычисляем коэффициенты $k_0^{(4)}_{y_1} = y_2 = 2.052658$, $k_0^{(4)}_{y_2} = 0.8 - 2 \cdot 1.60 \cdot 1.202760 + 1.60 \cdot 2.052658 = 0.235421$. Записываем эти же значения в столбец Δf .

5. Используя найденные значения коэффициентов $k_0^{(1-4)}_{y_1}$ и $k_0^{(1-4)}_{y_2}$ вычисляем $\Delta y_1 = \frac{1}{6} (k_0^{(1)}_{y_1} + 2k_0^{(2)}_{y_1} + 2k_0^{(3)}_{y_1} + k_0^{(4)}_{y_1}) = \frac{1}{6} (2.000000 + 4.080000 + 4.055200 + 2.052658) = 2.031310$.

Аналогично $\Delta y_2 = 0.532096$.

6. Для $i = 1$ находим $x_1 = x_0 + h = 1.60$, $y_1(x_1) = y_1(x_0) + \Delta y_1 = 1 + 2.031310 = 3.031310$, $y_2(x_1) = y_2(x_0) + \Delta y_2 = 2 + 0.532096 = 2.532096$. Далее все вычисления повторяются до достижения $x = x_n = 1.8$. Результаты помещены в следующую таблицу:

i	x	$(y_1(x_i), y_2(x_i))$	$k_i^{(1-4)}$	Δf	$\Delta y = 1/6 \sum \alpha k$
0	1.50	1	2	2	
		2	0.8	0.8	
	1.55	1.100000	2.040000	4.080000	
		2.040000	0.552000	1.104000	
	1.55	1.102000	2.027600	4.055200	
		2.027690	0.526580	1.053160	
	1.60	1.202760	2.052658	2.052658	2.031310
		2.052658	0.235421	0.235421	0.532096

i	x	$(y_1(x_i), y_2(x_i))$	$k_i^{(1-4)}$	Δf	$\Delta y = 1/6 \sum \alpha k$
1	1.60	3.031310	2.532096	2.532096	
		2.532096	-4.864838	-4.864838	
	1.65	3.157915	2.288854	4.577708	
		2.288854	-5.844510	-11.689020	
	1.65	3.145753	1.996629	3.993258	
		1.996629	-6.286547	-12.573094	
	1.70	3.230973	1.903441	1.903441	2.167751
		1.903441	-6.949459	-6.949459	-6.012735
	2	5.199061	-3.480639	-3.480639	
		-3.480639	-22.793893	-22.793893	
		5.025029	-4.620334	-9.240668	
		-4.620334	-24.642168	-49.284336	
		4.968044	-4.712747	-9.425494	
		-4.712747	-24.835461	-49.670922	
	1.80	4.727786	-5.964185	-5.964185	-4.685164
		-5.964185	-26.955562	-26.955562	-24.784118
3	1.80	0.513897			
		-28.264811			

7.11. Лабораторная работа № 13. Численное интегрирование систем дифференциальных уравнений первого порядка

Задача Коши для отдельного дифференциального уравнения решается сравнительно редко. Чаще приходится интегрировать систему обыкновенных дифференциальных уравнений. Дифференциальные уравнения высших порядков также легко сводятся к системе уравнений. Если задано уравнение $y^{(n)} = f(x, y(x), y'(x), \dots, y^{(n-1)}(x))$ с начальными условиями $y(x_0) = y_{10}, y'(x_0) = y_{20}, \dots, y^{(n-1)}(x_0) = y_{n0}$, то стандартная замена переменных $y_1(x) = y(x), y_2(x) = y'(x), \dots, y_n(x) = y^{(n-1)}(x)$ приводит это уравнение к системе n дифференциальных уравнений первого порядка:

$$\begin{cases} y'_1(x) = y_2(x), \\ y'_2(x) = y_3(x) \\ \dots \\ y'_{n-1}(x) = y_n(x) \\ y'_n(x) = f(x, y_1(x), y_2(x), \dots, y_n(x)) \end{cases} \quad (7.11.1)$$

с начальными условиями $y_1(x_0) = y_{10}, y_2(x_0) = y_{20}, \dots, y_n(x_0) = y_{n0}$.

Интегрирование систем дифференциальных уравнений в пакете Mathcad проводится теми же функциями, которые описаны в предыдущей лабораторной работе, поскольку задача Коши для уравнений сводится при использовании этих же функций к решению задачи Коши для систем.

Численное решение этой задачи состоит в построении таблицы приближенных значений $y_{i,1}, y_{i,2}, \dots, y_{i,k}$, $i = 1, 2, \dots, n$ решения $y_1(x), y_2(x), \dots, y_n(x)$ на отрезке $[x_0, x_k]$ в узлах сетки x_0, x_1, \dots, x_k . Пусть

$$\begin{aligned} \bar{Y}(x) &= (y_1(x), y_2(x), \dots, y_n(x))^T, \\ \bar{Y}_0 &= (y_{10}, y_{20}, \dots, y_{n0})^T \\ \bar{Y}'(x) &= (y'_1(x), y'_2(x), \dots, y'_n(x))^T \\ \bar{F}(x, \bar{Y}) &= (f_1(x, y_1, \dots, y_n), f_2(x, y_1, \dots, y_n), \dots, f_n(x, y_1, \dots, y_n))^T, \end{aligned}$$

где $\bar{Y}(x)$ - решение системы, \bar{Y}_0 - вектор начальных условий, $\bar{F}(x, \bar{Y})$ - вектор правых частей системы. Тогда исходная система дифференциальных уравнений первого порядка (7.11.1) в векторной форме перепишется в виде

$$\begin{cases} \bar{Y}' = \bar{F}(x, \bar{Y}), \\ \bar{Y}(x_0) = \bar{Y}_0. \end{cases} \quad (7.11.2)$$

Рассмотрим пример. Пусть $x \in [1, 5]$,

$$\begin{cases} y'_1 = \sin(xy_1 y_2 y_3), & y_1(1) = 0, \\ y'_2 = -\cos(xy_1 + y_3), & y_2(1) = -0.3, \\ y'_3 = e^{-(y_1 + y_2 + y_3)}, & y_3(1) = 1. \end{cases}$$

Воспользуемся всеми встроенными программами интегрирования систем обыкновенных дифференциальных уравнений, которыми располагает система Mathcad. Введем программу вычислений.

```
ORIGIN := 1 a := 1 b := 5
y :=  $\begin{pmatrix} 0 \\ 0.3 \\ 1 \end{pmatrix}$  f(x,y) :=  $\begin{pmatrix} \sin(x \cdot y_1 \cdot y_2 \cdot y_3) \\ -\cos(x \cdot y_1 + y_3) \\ \exp(-(y_1 + y_2 + y_3)) \end{pmatrix}$ 
z := rkfixed(y,a,b,10,f)
z1 := Rkadapt(y,a,b,10,f)
z2 := Bulstoer(y,a,b,10,f)
```

Последняя функция **Bulstoer** решает задачу Коши на отрезке $[a, b]$ методом Булирша – Штера. Он является методом рациональной экстраполяции, главным достоинством которого является то, что для достижения высокой точности не требуется многократного пересчета правых частей интегрируемых уравнений. Это особенно удобно, когда правые части уравнений сложны. Основная идея метода рациональной экстраполяции заключается в следующем. Сначала находится некоторое приближенное решение рассматриваемых уравнений в точках h_k , например, по методу Эйлера; затем рассчитывается улучшенное приближение путем экстраполяции рациональными функциями, например, многочленами по специальным вычислительным схемам.

Для сложных систем, насчитывающих десятки дифференциальных уравнений, основной выигрыш при применении метода Булирша – Штера, помимо точности, заключается в заметном сокращении времени вычислений.

Приведем результаты вычислений по всем трем используемым программам и графики решений ($z2$ и графики см. на следующей странице):

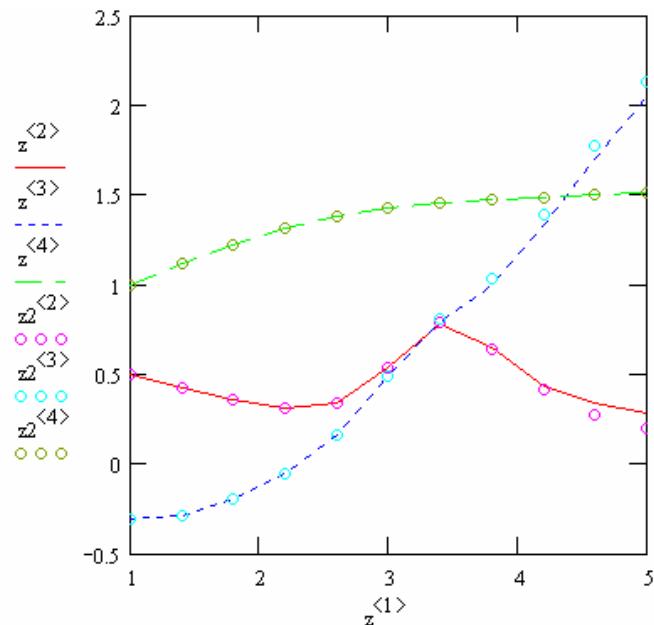
	1	2	3	4
1	1	0.5	-0.3	1
2	1.4	0.4299059	-0.2817295	1.1172544
3	1.8	0.3595994	-0.1913401	1.2237941
4	2.2	0.3182454	-0.0471412	1.3147827
5	2.6	0.3416382	0.1628289	1.3863529
6	3	0.5427708	0.4928312	1.4338279
7	3.4	0.7797098	0.7984027	1.4579901
8	3.8	0.654266	0.9981075	1.4756457
9	4.2	0.4412921	1.3388277	1.4921683
10	4.6	0.3400564	1.7065483	1.5058655
11	5	0.291659	2.0478717	1.515831

	1	2	3	4
1	1	0.5	-0.3	1
2	1.4	0.4299597	-0.2817752	1.117245
3	1.8	0.3596558	-0.1913855	1.2237752
4	2.2	0.3182623	-0.0471173	1.3147486
5	2.6	0.3416142	0.1631264	1.3862756
6	3	0.5453174	0.4952548	1.4336313
7	3.4	0.7957512	0.8095403	1.4576682
8	3.8	0.6424303	1.0362596	1.4753814
9	4.2	0.4197688	1.3897379	1.4914458
10	4.6	0.2752556	1.7809718	1.5045029
11	5	0.2032278	2.1302184	1.5143829

Видно, что функция **rkfixed** по y_1 и y_2 на конце интервала интегрирования дает уже неудовлетворительный результат, что объясняется слишком большим шагом интегрирования $h = 0.5$.

Рассмотрим еще один метод интегрирования так называемых «жестких» систем дифференциальных уравнений. Эти системы характерны тем, что, несмотря на медленное изменение функций, определяющих решение, расчеты приходится вести с очень мелким шагом. Все попытки увеличить шаг и тем самым уменьшить время решения задачи приводят лишь к катастрофически большому росту погрешности. Термин «жесткий» происходит из механики, где численное решение некоторых систем дифференциальных уравнений требует разного шага интегрирования по разным искомым функциям. Численное решение «жестких» задач требует применения специальных неявных методов.

	1	2	3	4
1	1	0.5	-0.3	1
2	1.4	0.4299588	-0.2817736	1.117245
3	1.8	0.3596549	-0.1913832	1.2237751
4	2.2	0.3182619	-0.0471155	1.3147486
5	2.6	0.3416133	0.163125	1.3862764
6	3	0.5452961	0.4952358	1.4336345
7	3.4	0.7952794	0.8092927	1.4576658
8	3.8	0.6422105	1.0361777	1.475382
9	4.2	0.4198404	1.3897118	1.4914485
10	4.6	0.2754648	1.7809081	1.5045058
11	5	0.2030736	2.1302164	1.5143854



В пакете Mathcad подобные системы решаются с помощью функций **Stiffb** и **Stiffr**, которые имеют те же параметры, что и функция **rkfixed**. Кроме того, задается информация о скорости изменения вектора правых частей уравнений (7.11.2), то есть матрица Якоби правых частей:

$$J(x, y) = \begin{pmatrix} \frac{\partial f_1(x, y)}{\partial x} & \frac{\partial f_1(x, y)}{\partial y_1} & \dots & \frac{\partial f_1(x, y)}{\partial y_n} \\ \frac{\partial f_2(x, y)}{\partial x} & \frac{\partial f_2(x, y)}{\partial y_1} & \dots & \frac{\partial f_2(x, y)}{\partial y_n} \\ \vdots & \vdots & \ddots & \vdots \\ \frac{\partial f_n(x, y)}{\partial x} & \frac{\partial f_n(x, y)}{\partial y_1} & \dots & \frac{\partial f_n(x, y)}{\partial y_n} \end{pmatrix}$$

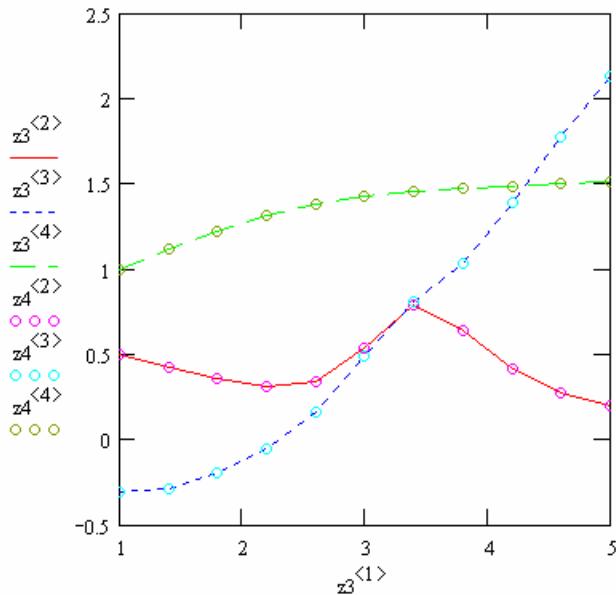
Введем следующую часть программы:

$$J(x, y) := \begin{bmatrix} y_1 \cdot y_2 \cdot y_3 \cdot \cos(x \cdot y_1 \cdot y_2 \cdot y_3) & x \cdot y_2 \cdot y_3 \cdot \cos(x \cdot y_1 \cdot y_2 \cdot y_3) & y_1 \cdot x \cdot y_3 \cdot \cos(x \cdot y_1 \cdot y_2 \cdot y_3) & y_1 \cdot y_2 \cdot x \cdot \cos(x \cdot y_1 \cdot y_2 \cdot y_3) \\ y_1 \cdot \sin(x \cdot y_1 + y_3) & x \cdot \sin(x \cdot y_1 + y_3) & 0 & \sin(x \cdot y_1 + y_3) \\ 0 & -\exp[-(y_1 + y_2 + y_3)] & -\exp[-(y_1 + y_2 + y_3)] & -\exp[-(y_1 + y_2 + y_3)] \end{bmatrix}$$

$$z3 := \text{Stiffb}(y, a, b, 10, f, J) \quad z4 := \text{Stiffr}(y, a, b, 10, f, J)$$

	1	2	3	4
1	1	0.5	-0.3	1
2	1.4	0.4299624	-0.2817799	1.1172456
3	1.8	0.3596568	-0.1913938	1.2237774
4	2.2	0.3182542	-0.0471335	1.3147544
5	2.6	0.3415641	0.1630618	1.3862917
6	3	0.5451887	0.4951536	1.4336536
7	3.4	0.7959177	0.8095354	1.4576888
8	3.8	0.6425095	1.0361398	1.4754012
9	4.2	0.4197355	1.3895633	1.4914466
10	4.6	0.2750179	1.780831	1.5045243
11	5	0.2029673	2.1301055	1.5144057

	1	2	3	4
1	1	0.5	-0.3	1
2	1.4	0.4299592	-0.2817723	1.1172448
3	1.8	0.3596538	-0.1913771	1.2237745
4	2.2	0.3182586	-0.0471018	1.3147475
5	2.6	0.3415982	0.1631487	1.3862768
6	3	0.5452929	0.4952999	1.4336331
7	3.4	0.7957205	0.8096313	1.4576755
8	3.8	0.6422773	1.0364828	1.4753868
9	4.2	0.4195632	1.3899233	1.4914466
10	4.6	0.2749778	1.7810319	1.5045008
11	5	0.2028309	2.1302548	1.5143797



Функция **Stiffb** использует алгоритм Булирша – Штера, функция **Stiffr** – алгоритм Розенброка, подробно описанный в специальной литературе.

Приведем, наконец, текст подпрограммы, реализующий метод Рунге – Кутты четвертого порядка с постоянным шагом для систем дифференциальных уравнений с параметрами, полностью аналогичными параметрам функции **rkfixed**:

```

RGKsyst(y,a,b,n,f) := | h<- (b-a)
                         |   n
                         | z1,1<-a
                         | k<-rows(y)
                         | for i< 1..k
                         |   z1,i+1<-y_i
                         | for i< 2..n + 1
                         |   | h1<-z1,-1,1 + h
                         |   |   2
                         |   | k1<-f(z1,-1,1,y)
                         |   | yT<-k1
                         |   | for j< 1..k

```

```

y1<- $(yT^{<j>})_1$ 
dj<-zi-1,j+1 +  $\frac{h}{2} \cdot y1$ 
k2<-f(h1, d)
yT<-k2T
for j ε 1.. k
y1<- $(yT^{<j>})_1$ 
dj<-zi-1,j+1 +  $\frac{h}{2} \cdot y1$ 
k3<-f(h1, d)
yT<-k3T
for j ε 1.. k
y1<- $(yT^{<j>})_1$ 
dj<-zi-1,j+1 + h · y1
k4<-f $\left(h1 + \frac{h}{2}, d\right)$ 
for j ε 1.. k
dj<- $\frac{1}{6} \cdot [k1_j + 2 \cdot (k2_j + k3_j) + k4_j]$ 
zi,j+1<-zi-1,j+1 + h · dj
yj<-zi,j+1
zi,1<-h1 +  $\frac{h}{2}$ 

```

z

$z5 := \text{RGKsyst}(y, a, b, 10, f)$

	1	2	3	4
1	1	0.5	-0.3	1
2	1.4	0.4299059	-0.2817295	1.1172544
3	1.8	0.3595994	-0.1913401	1.2237941
4	2.2	0.3182454	-0.0471412	1.3147827
5	2.6	0.3416382	0.1628289	1.3863529
6	3	0.5427708	0.4928312	1.4338279
7	3.4	0.7797098	0.7984027	1.4579901
8	3.8	0.654266	0.9981075	1.4756457
9	4.2	0.4412921	1.3388277	1.4921683
10	4.6	0.3400564	1.7065483	1.5058655
11	5	0.291659	2.0478717	1.515831

	1	2	3	4
1	1	0.5	-0.3	1
2	1.4	0.4299059	-0.2817295	1.1172544
3	1.8	0.3595994	-0.1913401	1.2237941
4	2.2	0.3182454	-0.0471412	1.3147827
5	2.6	0.3416382	0.1628289	1.3863529
6	3	0.5427708	0.4928312	1.4338279
7	3.4	0.7797098	0.7984027	1.4579901
8	3.8	0.654266	0.9981075	1.4756457
9	4.2	0.4412921	1.3388277	1.4921683
10	4.6	0.3400564	1.7065483	1.5058655
11	5	0.291659	2.0478717	1.515831

Сравнение матриц z и $z5$ показывает, что результаты расчета по подпрограммам **rkfixed** и **RGKsyst** практически одинаковы.

Задание № 1. С помощью любой из разобранных в лабораторной работе подпрограмм решить задачу Коши для системы дифференциальных уравнений на заданном отрезке:

1. $\begin{cases} y'_1 = \frac{1}{2 + \sin(y_1 y_3)}, & y_1(0) = 1, \\ y'_2 = \frac{3}{3 + \cos(y_1 y_2 y_3)}, & x \in [0, 2], y_2(0) = 0, \\ y'_3 = \frac{1}{2 + e^{-y_1 y_2}}, & y_3(0) = 0. \end{cases}$
2. $\begin{cases} y'_1 = \sqrt[3]{x y_1 y_3 + y_2}, & y_1(0.5) = -2, \\ y'_2 = \sqrt[3]{y_1 + y_2 + y_3}, & x \in [0.5, 0.1], y_2(0.5) = -1, \\ y'_3 = y_2 e^{\sin y_1}, & y_3(0.5) = 0.5. \end{cases}$
3. $\begin{cases} y'_1 = x e^{-(y_1 + y_2)}, & y_1(0) = -1, \\ y'_2 = y_1 y_2 y_3, & x \in [0, 3], y_2(0) = 0.3, \\ y'_3 = \sin\left(\frac{x}{y_1^2 + 1}\right) \cdot y_2, & y_3(0) = 0.45. \end{cases}$
4. $\begin{cases} y'_1 = x y_1 + x^2 y_2 y_3, & y_1(2) = 0, \\ y'_2 = -y_1 - y_2 + y_3^2, & x \in [2, 5], y_2(2) = 0, \\ y'_3 = \ln(1 + y_1^2 y_2^2), & y_3(2) = 1. \end{cases}$
5. $\begin{cases} y'_1 = \sqrt{1 + y_1^2 + y_2^2}, & y_1(-1) = 1, \\ y'_2 = \sin(xy_1 y_2 y_3), & x \in [-1, 1], y_2(-1) = -1, \\ y'_3 = -\cos(xy_1 + y_3), & y_3(-1) = -2. \end{cases}$
6. $\begin{cases} y'_1 = e^{-(y_1 + y_2 + y_3)}, & y_1(0) = -1, \\ y'_2 = \operatorname{arctg}(xy_1 y_3), & x \in [0, 3], y_2(0) = -1, \\ y'_3 = \sin(\operatorname{arctg}(y_1 y_3)), & y_3(0) = -2. \end{cases}$
7. $\begin{cases} y'_1 = 2y_1 + y_2 y_3, & y_1(0) = 0.2, \\ y'_2 = y_1 y_2 y_3 + y_1^2, & x \in [0, 4], y_2(0) = 0, \\ y'_3 = x y_1 + y_2 - y_3^2, & y_3(0) = 0. \end{cases}$
8. $\begin{cases} y'_1 = \sin(y_1 y_2 + y_3), & y_1(0.3) = -0.4, \\ y'_2 = \operatorname{sh}(xy_1 + y_2 y_3), & x \in [0.3, 2], y_2(0.3) = -0.4, \\ y'_3 = \operatorname{ch}(xy_1 y_2 + y_3), & y_3(0.3) = 1. \end{cases}$
9. $\begin{cases} y'_1 = x e^{-(y_1 + y_2)}, & y_1(0) = 0, \\ y'_2 = y_1 y_2 y_3, & x \in [0, 3], y_2(0) = 1, \\ y'_3 = x^2 + y_2^2, & y_3(0) = -3. \end{cases}$
10. $\begin{cases} y'_1 = \cos(y_1 y_2 y_3), & y_1(2) = 0.5, \\ y'_2 = y_1 + y_2 + y_3, & x \in [2, 5], y_2(2) = 1, \\ y'_3 = \sin(y_1 + y_3), & y_3(2) = -0.3. \end{cases}$
11. $\begin{cases} y'_1 = \sin(y_1 + y_3), & y_1(0) = 0, \\ y'_2 = \cos(y_1 y_2 y_3), & x \in [0, 3], y_2(0) = 1, \\ y'_3 = y_1 + y_2 + y_3, & y_3(0) = -1. \end{cases}$

12. $\begin{cases} y'_1 = \ln(1 + y_1^2 y_2^2), & y_1(0) = 1, \\ y'_2 = xy_1 + x^2 y_2 y_3, & x \in [0, 2], y_2(0) = 1, \\ y'_3 = -y_1 - y_2 - y_3^2, & y_3(0) = 0. \end{cases}$
13. $\begin{cases} y'_1 = 2y_1 + y_2 y_3, & y_1(-2) = -1, \\ y'_2 = y_1 y_2 y_3 + y_1^2, & x \in [-2, 2], y_2(-2) = -5, \\ y'_3 = \sqrt{1 + y_1^2 + y_2^2}, & y_3(-2) = 1. \end{cases}$
14. $\begin{cases} y'_1 = y_1 y_2 y_3, & y_1(0) = 0, \\ y'_2 = x^2 + y_2^2, & x \in [0, 2], y_2(0) = -0.5, \\ y'_3 = \ln(x^2 + y_1^2), & y_3(0) = 0.5. \end{cases}$
15. $\begin{cases} y'_1 = \sin(y_1 + y_3), & y_1(-1) = 1, \\ y'_2 = \cos(y_1 y_2 y_3), & x \in [-1, 1], y_2(-1) = -1, \\ y'_3 = -\cos(xy_1 + y_3), & y_3(-1) = -2. \end{cases}$
16. $\begin{cases} y'_1 = \operatorname{arctg} \frac{1}{1+x^2+y_2^2}, & y_1(1) = -1, \\ y'_2 = \frac{1}{1+x^2+y_2^2}, & x \in [1, 3], y_2(1) = 0, \\ y'_3 = \frac{3}{1+y_1^2+y_2^2+y_3^2}, & y_3(1) = 1. \end{cases}$
17. $\begin{cases} y'_1 = \sin \left(\frac{x}{1+y_1^2} \right) \cdot y_2, & y_1(0) = 1, \\ y'_2 = \sqrt[3]{xy_1 y_3 + y_2}, & x \in [0, 3], y_2(0) = 1, \\ y'_3 = \sqrt[3]{y_1 + y_2 + y_3}, & y_3(0) = 1. \end{cases}$
18. $\begin{cases} y'_1 = \sin(y_1 y_2 y_3), & y_1(1) = 0, \\ y'_2 = \ln(x^2 + y_3^2), & x \in [1, 4], y_2(1) = 0, \\ y'_3 = e^{-y_1 y_2 y_3}, & y_3(1) = 0. \end{cases}$
19. $\begin{cases} y'_1 = y_2 e^{\sin y_1}, & y_1(0) = 0, \\ y'_2 = y_3 e^{\cos(xy_2)}, & x \in [0, 2], y_2(0) = 0, \\ y'_3 = e^{\sin(y_1 y_2 y_3)}, & y_3(0) = 0. \end{cases}$
20. $\begin{cases} y'_1 = \operatorname{sh}(xy_1 + y_2 y_3), & y_1(1) = 0, \\ y'_2 = \operatorname{ch}(xy_1 y_2 + y_3), & x \in [1, 4], y_2(1) = -1, \\ y'_3 = \sin(y_1 y_2 + y_3), & y_3(1) = 0.4. \end{cases}$
21. $\begin{cases} y'_1 = xy_1 + y_2 - y_3^2, & y_1(-1) = -1, \\ y'_2 = 2y_1 + y_2 y_3, & x \in [-1, 3], y_2(-1) = 1, \\ y'_3 = y_1 y_2 y_3 + y_1^2, & y_3(-1) = 2. \end{cases}$
22. $\begin{cases} y'_1 = 2y_1 + y_2 y_3, & y_1(-1) = -1, \\ y'_2 = y_1 y_2 y_3 + y_1^2, & x \in [-1, 2], y_2(-1) = -5, \\ y'_3 = \sqrt{1 + y_1^2 + y_2^2}, & y_3(-1) = 1. \end{cases}$

23. $\begin{cases} y'_1 = \ln(x + y_1^2 + y_3^2), & y_1(1) = 0, \\ y'_2 = \operatorname{sh}(xy_1y_2) + y_3, & y_2(1) = 1, \\ y'_3 = y_1 + 3y_2 - xy_3^2, & y_3(1) = -1. \end{cases}$
24. $\begin{cases} y'_1 = xy_1 + x^2y_2y_3, & y_1(0) = -0.3, \\ y'_2 = -y_1 - y_2 + y_3^2, & y_2(0) = 0.5, \\ y'_3 = xe^{-(y_1+y_2)}, & y_3(0) = 1. \end{cases}$
25. $\begin{cases} y'_1 = \operatorname{arctg}(xy_1y_3), & y_1(1) = 0, \\ y'_2 = \sin(\operatorname{arctg}(y_1y_3)), & y_2(1) = -0.3, \\ y'_3 = e^{-(y_1+y_2+y_3)}, & y_3(1) = 1. \end{cases}$
26. $\begin{cases} y'_1 = x(y_1 + y_2 \sin y_3), & y_1(1) = 0, \\ y'_2 = x \cos(y_2 + y_3), & y_2(1) = -0.5, \\ y'_3 = \ln(x^2 + y_1^2), & y_3(1) = 0.5. \end{cases}$
27. $\begin{cases} y'_1 = y_1y_2y_3, & y_1(0) = 0, \\ y'_2 = x^2 + y_1^2, & y_2(0) = -0.5, \\ y'_3 = \ln(x^2 + y_2^2), & y_3(0) = 0.5. \end{cases}$
28. $\begin{cases} y'_1 = 2y_1 + y_2y_3, & y_1(0) = 0.2, \\ y'_2 = y_1y_2y_3 + y_1^2, & y_2(0) = 0, \\ y'_3 = xy_1 + y_2 - y_3^2, & y_3(0) = 0. \end{cases}$
29. $\begin{cases} y'_1 = \ln(x^2 + y_1^2), & y_1(-1) = 0, \\ y'_2 = \operatorname{arctg}(xy_1y_3), & y_2(-1) = -1, \\ y'_3 = \sin(\operatorname{arctg}(y_1y_3)), & y_3(-1) = 1. \end{cases}$
30. $\begin{cases} y'_1 = \sin y_1 \cdot \cos^3 y_2, & y_1(-1) = 0, \\ y'_2 = x \cos y_1 \cdot \cos y_3, & y_2(-1) = 0, \\ y'_3 = \sin(x - y_2 - y_3), & y_3(-1) = 0. \end{cases}$

8. ДИФФЕРЕНЦИАЛЬНЫЕ УРАВНЕНИЯ С ЧАСТНЫМИ ПРОИЗВОДНЫМИ (УРАВНЕНИЯ МАТЕМАТИЧЕСКОЙ ФИЗИКИ)

8.1. Классификация уравнений математической физики

Дифференциальным уравнением с частными производными называется уравнение относительно неизвестной функции $u(x, y)$ двух или более независимых переменных, которое содержит частные производные этой функции.

Порядком дифференциального уравнения называется наибольший порядок производных, входящих в это уравнение.

Важнейшее значение с точки зрения приложений в физике и технике имеют уравнения с частными производными второго порядка, поэтому основное внимание будет уделено изучению именно таких уравнений. В общем виде дифференциальное уравнение второго порядка относительно функции двух независимых переменных $u(x, y)$ записывается следующим образом:

$$F\left(x, y, u, \frac{\partial u}{\partial x}, \frac{\partial u}{\partial y}, \frac{\partial^2 u}{\partial x^2}, \frac{\partial^2 u}{\partial y^2}, \frac{\partial^2 u}{\partial x \partial y}\right) = 0, \quad (8.1.1)$$

где F - заданная функция восьми аргументов.

Далее будут рассматриваться, в основном, уравнения более простого вида, чем (8.1.1), а именно, линейные уравнения второго порядка

$$a(x, y) \frac{\partial^2 u}{\partial x^2} + 2b(x, y) \frac{\partial^2 u}{\partial x \partial y} + c(x, y) \frac{\partial^2 u}{\partial y^2} + d(x, y) \frac{\partial u}{\partial x} + e(x, y) \frac{\partial u}{\partial y} + g(x, y)u = f(x, y), \quad (8.1.2)$$

где коэффициенты $a(x, y), b(x, y), \dots, g(x, y)$ и правая часть $f(x, y)$ - функции, не зависящие от u , которые заданы и непрерывны в некоторой области D , называемой областью определения дифференциального уравнения.

Если все коэффициенты уравнения (8.1.2) не зависят от x и y , то оно называется уравнением с постоянными коэффициентами, если же $f(x, y) = 0$, уравнение называется однородным.

Естественно возникает вопрос о существовании наиболее компактной формы записи уравнения (8.1.2). Оказывается, этого можно добиться надлежащей заменой переменных. Перепишем уравнение (8.1.2) в следующем виде:

$$a_{11}u''_{xx} + 2a_{12}u''_{xy} + a_{22}u''_{yy} + F_1(x, y, u(x, y), u'_x, u'_y) = 0, \quad (8.1.3)$$

где $a_{11} = a_{11}(x, y)$, $a_{12} = a_{12}(x, y)$, $a_{22} = a_{22}(x, y)$, $f(x, y) = 0$. Это уравнение называется линейным относительно старших производных. Сделаем в нем замену переменных $\varphi = \varphi(x, y)$ и $\psi = \psi(x, y)$, обеспечивающую взаимно однозначное соответствие между x, y и φ, ψ . Тогда

$$\begin{cases} \varphi = \varphi(x, y), \\ \psi = \psi(x, y), \end{cases} \Leftrightarrow \begin{cases} x = x(\varphi, \psi), \\ y = y(\varphi, \psi) \end{cases} \quad \text{и} \quad D = \begin{vmatrix} \varphi'_x & \psi'_x \\ \varphi'_y & \psi'_y \end{vmatrix} \neq 0. \quad \text{Так как } u = u(x, y), \quad \text{то}$$

$u'_x = [u(\varphi(x, y), \psi(x, y))]'_x = \frac{\partial u}{\partial \varphi} \cdot \frac{\partial \varphi}{\partial x} + \frac{\partial u}{\partial \psi} \cdot \frac{\partial \psi}{\partial x}$, $u'_y = \frac{\partial u}{\partial \varphi} \cdot \frac{\partial \varphi}{\partial y} + \frac{\partial u}{\partial \psi} \cdot \frac{\partial \psi}{\partial y}$. Найдем вторую производную по x , расписав подробно получающееся выражение:

$$\begin{aligned} u''_{xx} &= \left(\frac{\partial u}{\partial \varphi} \cdot \frac{\partial \varphi}{\partial x} + \frac{\partial u}{\partial \psi} \cdot \frac{\partial \psi}{\partial x} \right)'_x = \left(\frac{\partial u}{\partial \varphi} \cdot \frac{\partial \varphi}{\partial x} \right)'_x + \left(\frac{\partial u}{\partial \psi} \cdot \frac{\partial \psi}{\partial x} \right)'_x = \left(\frac{\partial u}{\partial \varphi} \right)'_x \cdot \frac{\partial \varphi}{\partial x} + \frac{\partial u}{\partial \varphi} \cdot \frac{\partial^2 \varphi}{\partial x^2} + \\ &+ \left(\frac{\partial u}{\partial \psi} \right)'_x \cdot \frac{\partial \psi}{\partial x} + \frac{\partial u}{\partial \psi} \cdot \frac{\partial^2 \psi}{\partial x^2} = \left\{ \begin{array}{l} \left(\frac{\partial u}{\partial \varphi} \right)'_x = \left[\frac{\partial u(\varphi(x, y), \psi(x, y))}{\partial \varphi} \right]'_x = \\ = \frac{\partial^2 u}{\partial \varphi^2} \cdot \frac{\partial \varphi}{\partial x} + \frac{\partial^2 u}{\partial \varphi \partial \psi} \cdot \frac{\partial \psi}{\partial x} \end{array} \right\} = \left(\frac{\partial^2 u}{\partial \varphi^2} \cdot \frac{\partial \varphi}{\partial x} + \right. \\ &\left. + \frac{\partial^2 u}{\partial \varphi \partial \psi} \cdot \frac{\partial \psi}{\partial x} \right) \cdot \frac{\partial \varphi}{\partial x} + \frac{\partial u}{\partial \varphi} \cdot \frac{\partial^2 \varphi}{\partial x^2} + \left(\frac{\partial^2 u}{\partial \varphi \partial \psi} \cdot \frac{\partial \varphi}{\partial x} + \frac{\partial^2 u}{\partial \psi^2} \cdot \frac{\partial \psi}{\partial x} \right) \cdot \frac{\partial \psi}{\partial x} + \frac{\partial u}{\partial \psi} \cdot \frac{\partial^2 \psi}{\partial x^2} = \\ &= \frac{\partial^2 u}{\partial \varphi^2} \cdot \left(\frac{\partial \varphi}{\partial x} \right)^2 + \frac{\partial^2 u}{\partial \varphi \partial \psi} \cdot \frac{\partial \varphi}{\partial x} \cdot \frac{\partial \psi}{\partial x} + \frac{\partial u}{\partial \varphi} \cdot \frac{\partial^2 \varphi}{\partial x^2} + \frac{\partial^2 u}{\partial \varphi \partial \psi} \cdot \frac{\partial \varphi}{\partial x} \cdot \frac{\partial \psi}{\partial x} + \frac{\partial^2 u}{\partial \psi^2} \cdot \left(\frac{\partial \psi}{\partial x} \right)^2 + \\ &+ \frac{\partial u}{\partial \psi} \cdot \frac{\partial^2 \psi}{\partial x^2} = \frac{\partial^2 u}{\partial \varphi^2} \cdot \left(\frac{\partial \varphi}{\partial x} \right)^2 + \frac{\partial^2 u}{\partial \psi^2} \cdot \left(\frac{\partial \psi}{\partial x} \right)^2 + 2 \frac{\partial^2 u}{\partial \varphi \partial \psi} \cdot \frac{\partial \varphi}{\partial x} \cdot \frac{\partial \psi}{\partial x} + \frac{\partial u}{\partial \varphi} \cdot \frac{\partial^2 \varphi}{\partial x^2} + \frac{\partial u}{\partial \psi} \cdot \frac{\partial^2 \psi}{\partial x^2} = \\ &= u''_{\varphi\varphi} \cdot (\varphi'_x)^2 + u''_{\psi\psi} \cdot (\psi'_x)^2 + 2u''_{\varphi\psi} \varphi'_x \psi'_x + u'_{\varphi} \varphi''_{xx} + u'_{\psi} \psi''_{xx}. \end{aligned}$$

Аналогично можно записать и $u''_{yy} = u''_{\varphi\varphi} \cdot (\varphi'_y)^2 + u''_{\psi\psi} \cdot (\psi'_y)^2 + 2u''_{\varphi\psi} \varphi'_y \psi'_y + u'_{\varphi} \varphi''_{yy} + u'_{\psi} \psi''_{yy}$, $u''_{xy} = u''_{\varphi\varphi} \varphi'_x \varphi'_y + u''_{\psi\psi} \psi'_x \psi'_y + u''_{\varphi\psi} (\varphi'_x \psi'_y + \varphi'_y \psi'_x) + u'_{\varphi} \varphi''_{xy} + u'_{\psi} \psi''_{xy}$. Подставляя эти значения в уравнение (8.1.3), получим

$$\begin{aligned} a_{11} &\left(u''_{\varphi\varphi} (\varphi'_x)^2 + u''_{\psi\psi} (\psi'_x)^2 + 2u''_{\varphi\psi} \varphi'_x \psi'_x + u'_{\varphi} \varphi''_{xx} + u'_{\psi} \psi''_{xx} \right) + 2a_{12} \left(u''_{\varphi\varphi} \varphi'_x \varphi'_y + u''_{\psi\psi} \psi'_x \psi'_y + u''_{\varphi\psi} (\varphi'_x \psi'_y + \varphi'_y \psi'_x) + u'_{\varphi} \varphi''_{xy} + u'_{\psi} \psi''_{xy} \right) + \\ &+ a_{22} \left(u''_{\varphi\varphi} (\varphi'_y)^2 + u''_{\psi\psi} (\psi'_y)^2 + 2u''_{\varphi\psi} \varphi'_y \psi'_y + u'_{\varphi} \varphi''_{yy} + u'_{\psi} \psi''_{yy} \right) + \\ &+ F_1(\varphi, \psi, u(\varphi, \psi), u'_{\varphi} \varphi'_x + u'_{\psi} \psi'_x, u'_{\varphi} \varphi'_y + u'_{\psi} \psi'_y) = 0 \end{aligned}$$

или $\tilde{a}_{11}u_{\varphi\varphi}'' + 2\tilde{a}_{12}u_{\varphi\psi}'' + \tilde{a}_{22}u_{\psi\psi}'' + \tilde{F} = 0$, где $\tilde{F} = F_1(\varphi, \psi, u, u'_\varphi, u'_\psi)$, а

$$\begin{cases} \tilde{a}_{11} = a_{11}(\varphi'_x)^2 + 2a_{12}\varphi'_x\varphi'_y + a_{22}(\varphi'_y)^2, \\ \tilde{a}_{12} = a_{11}\varphi'_x\psi'_x + a_{12}(\varphi'_x\psi'_x + \varphi'_y\psi'_y) + a_{22}\varphi'_y\psi'_y, \\ \tilde{a}_{22} = a_{11}(\psi'_x)^2 + 2a_{12}\psi'_x\psi'_y + a_{22}(\psi'_y)^2. \end{cases}$$

Естественно, для упрощения уравнения $\tilde{a}_{11}u_{\varphi\varphi}'' + 2\tilde{a}_{12}u_{\varphi\psi}'' + \tilde{a}_{22}u_{\psi\psi}'' + \tilde{F} = 0$ в новых переменных φ и ψ надо так подобрать эти переменные, чтобы хотя бы некоторые члены рассматриваемого уравнения упрощались или вообще обнулялись. Например, можно выбрать φ и ψ так, чтобы $\tilde{a}_{11} = 0$. Для этого рассмотрим вспомогательное уравнение $a_{11}z_x'^2 + 2a_{12}z_x'z_y' + a_{22}z_y'^2 = 0$, где $z = z(x, y)$. Пусть $\varphi(x, y)$ - какое-нибудь частное решение этого уравнения. Если теперь выбрать $z = \varphi$, то $\tilde{a}_{11} = 0$. Аналогично, если $\psi(x, y)$ - другое частное решение этого же уравнения и $z = \psi$, то $\tilde{a}_{22} = 0$.

Теорема 8.1. Если $z = \varphi(x, y)$ является частным решением уравнения $a_{11}z_x'^2 + 2a_{12}z_x'z_y' + a_{22}z_y'^2 = 0$, то $\varphi(x, y) = C$ есть общий интеграл обыкновенного дифференциального уравнения

$$a_{11}dy^2 - 2a_{12}dxdy + a_{22}dx^2 = 0. \quad (8.1.4)$$

Наоборот, если $\varphi(x, y) = C$ - общий интеграл обыкновенного дифференциального уравнения (8.1.4), то функция $z = \varphi(x, y)$ удовлетворяет уравнению $a_{11}z_x'^2 + 2a_{12}z_x'z_y' + a_{22}z_y'^2 = 0$.

Решим уравнение (8.1.4), преобразовав его: $a_{11}\left(\frac{dy}{dx}\right)^2 - 2a_{12}\frac{dy}{dx} + a_{22} = 0$,

$$\left(\frac{dy}{dx}\right)_{1,2} = \frac{2a_{12} \pm \sqrt{4a_{12}^2 - 4a_{11}a_{22}}}{2a_{11}} = \frac{a_{12} \pm \sqrt{a_{12}^2 - a_{11}a_{22}}}{a_{11}}.$$

Таким образом, получим два обыкновенных дифференциальных уравнения:

$$\begin{cases} \frac{dy}{dx} = \frac{a_{12} + \sqrt{a_{12}^2 - a_{22}}}{a_{11}}, \\ \frac{dy}{dx} = \frac{a_{12} - \sqrt{a_{12}^2 - a_{22}}}{a_{11}}. \end{cases} \quad (8.1.5)$$

Знак подкоренного выражения определяет тип исходного уравнения (8.1.3) в частных производных.

1. Если в точке $M(x, y)$ $a_{12}^2 - a_{11}a_{22} > 0$, то уравнение $a_{11}u_{xx}'' + 2a_{12}u_{xy}'' + a_{22}u_{yy}'' + F = 0$ в этой точке называется уравнением гиперболического типа.

2. Если в точке $M(x, y)$ $a_{12}^2 - a_{11}a_{22} < 0$, то уравнение $a_{11}u_{xx}'' + 2a_{12}u_{xy}'' + a_{22}u_{yy}'' + F = 0$ в этой точке называется уравнением эллиптического типа.

3. Если в точке $M(x, y)$ $a_{12}^2 - a_{11}a_{22} = 0$, то уравнение $a_{11}u_{xx}'' + 2a_{12}u_{xy}'' + a_{22}u_{yy}'' + F = 0$ в этой точке называется уравнением параболического типа.

Так как $\tilde{a}_{12}^2 - \tilde{a}_{11}\tilde{a}_{22} = (a_{12}^2 - a_{11}a_{22})D$, то сказанное справедливо и для уравнения $\tilde{a}_{11}u_{\varphi\varphi}'' + 2\tilde{a}_{12}u_{\varphi\psi}'' + \tilde{a}_{22}u_{\psi\psi}'' + \tilde{F} = 0$.

Уравнение (8.1.4) называется характеристическим для уравнения (8.1.3), а интегралы уравнения (8.1.4) - характеристиками уравнения (8.1.3). Пусть в заданной области G уравнение (8.1.3) однотипно. Рассмотрим все три возможные случая подробнее.

1. Гиперболический тип. Два общих интеграла уравнения (8.1.3) $\varphi(x, y) = C$ и $\psi(x, y) = C$ определяют действительные семейства характеристик. Пусть $\varphi = \varphi(x, y)$ и $\psi = \psi(x, y)$ - новые переменные в уравнении (8.1.3). Тогда $2\tilde{a}_{12}u_{\varphi\psi}'' + \tilde{F} = 0$ или $u_{\varphi\psi}'' = -(\tilde{F}/2\tilde{a}_{12})$ - первая каноническая форма уравнения гиперболического типа. Если положить $\begin{cases} \varphi = \alpha + \beta, \\ \psi = \alpha - \beta, \end{cases} \Leftrightarrow \begin{cases} \alpha = \frac{\varphi + \psi}{2}, \\ \beta = \frac{\varphi - \psi}{2}, \end{cases}$ то $u_{\varphi}'' = u(\alpha + \beta, \alpha - \beta)'_{\varphi} = u'_{\alpha}\alpha'_{\varphi} + u'_{\beta}\beta'_{\varphi} = u'_{\alpha} \cdot \frac{1}{2} + u'_{\beta} \cdot \frac{1}{2} = \frac{1}{2}(u'_{\alpha} + u'_{\beta})$. Аналогично, $u_{\psi}'' = \frac{1}{2}(u'_{\alpha} - u'_{\beta})$ и $u_{\varphi\psi}'' = \frac{1}{4}(u''_{\alpha\alpha} - u''_{\beta\beta})$. В результате исходное уравнение примет вид $u''_{\alpha\alpha} - u''_{\beta\beta} = -(2\tilde{F}/\tilde{a}_{12}) = F_1$. Это вторая каноническая форма уравнения гиперболического типа.

2. Параболический тип. Для уравнения этого типа $a_{12}^2 - a_{11}a_{22} = 0$, имеется, следовательно, лишь один общий интеграл характеристического уравнения $a_{11}dy^2 - 2a_{12}dxdy + a_{22}dx^2 = 0$. Пусть этот интеграл $\varphi(x, y) = C$, положим $\varphi = \varphi(x, y)$, $\psi = \psi(x, y)$, где ψ - любая функция, не зависящая от φ . Тогда $\tilde{a}_{11} = a_{11}(\varphi'_x)^2 + 2a_{12}\varphi'_x\varphi'_y + a_{22}(\varphi'_y)^2 = (\sqrt{a_{11}}\varphi'_x + \sqrt{a_{22}}\varphi'_y)^2 = 0$, потому что при $a_{12}^2 - a_{11}a_{22} = 0$ $a_{12} = \sqrt{a_{11}a_{22}}$. Далее $\tilde{a}_{12} = a_{11}\varphi'_x\psi'_x + a_{12}(\varphi'_x\psi'_y + \varphi'_y\psi'_x) + a_{22}\varphi'_y\psi'_y = (\sqrt{a_{11}}\varphi'_x + \sqrt{a_{22}}\varphi'_y) \times (\sqrt{a_{11}}\psi'_x + \sqrt{a_{22}}\psi'_y) = 0$. Тогда исходное для этого случая уравнение $\tilde{a}_{11}u_{\varphi\varphi}'' + 2\tilde{a}_{12}u_{\varphi\psi}'' + \tilde{a}_{22}u_{\psi\psi}'' + \tilde{F} = 0$ обратится в следующее: $u_{\varphi\varphi}'' = -(\tilde{F}/\tilde{a}_{22}) = \Phi(\varphi, \psi, u, u'_{\varphi}, u'_{\psi})$. Это канонический вид уравнения параболического типа.

3. Эллиптический тип. Для этого типа уравнения $\begin{cases} \frac{dy}{dx} = \frac{a_{12} + \sqrt{a_{12}^2 - a_{22}}}{a_{11}}, \\ \frac{dy}{dx} = \frac{a_{12} - \sqrt{a_{12}^2 - a_{22}}}{a_{11}} \end{cases}$ имеют комплексные общие интегралы. Имеют комплексные общие интегралы. По свойству комплексной переменной, если $\varphi(x, y) = C$ - решение уравнения $a_{11}dy^2 - 2a_{12}dxdy + a_{22}dx^2 = 0$, то $\varphi^*(x, y)$ - комплексно-сопряженная функция также будет решением этого уравнения. Пусть $\varphi = \varphi(x, y)$, $\psi = \varphi^*(x, y)$, тогда уравнение эллиптического типа сводится к тому же виду, что и гиперболическое. Чтобы не иметь дела с комплексными переменными, введем новые переменные $\alpha = \frac{\varphi + \psi}{2}$, $\beta = \frac{\varphi - \psi}{2i}$, тогда $\varphi = \alpha + i\beta$, $\psi = \alpha - i\beta$. При этом предполагается, что a_{11}, a_{12}, a_{22} - аналитические функции.

$$\begin{aligned} & \text{Вычислим } a_{11}(\varphi'_x)^2 + 2a_{12}\varphi'_x\varphi'_y + a_{22}(\varphi'_y)^2 = 0 = a_{11}(\varphi'_{\alpha}\alpha'_x + \varphi'_{\beta}\beta'_x)^2 + 2a_{12}(\varphi'_{\alpha}\alpha'_x + \\ & + \varphi'_{\beta}\beta'_x)(\varphi'_{\alpha}\alpha'_y + \varphi'_{\beta}\beta'_y) + a_{22}(\varphi'_{\alpha}\alpha'_y + \varphi'_{\beta}\beta'_y)^2 = a_{11}(1 \cdot \alpha'_x + i \cdot \beta'_x)^2 + 2a_{12}(\alpha'_x + i \cdot \beta'_x) \cdot \\ & \cdot (\alpha'_y + i \cdot \beta'_y) + a_{22}(\alpha'_y + i \cdot \beta'_y)^2 = a_{11}(\alpha'^2_x + 2i\alpha'_x\beta'_x - \beta'^2_x) + 2a_{12}(\alpha'_x\alpha'_y + i\alpha'_y\beta'_x + i\alpha'_x\beta'_y - \\ & - \beta'_x\beta'_y) + a_{22}(\alpha'^2_y + 2i\alpha'_y\beta'_y - \beta'^2_y) = \underbrace{(a_{11}\alpha'^2_x + 2a_{12}\alpha'_x\alpha'_y + a_{22}\alpha'^2_y)}_{\tilde{a}_{11}} - \\ & - \underbrace{(a_{11}\beta'^2_x + 2a_{12}\beta'_x\beta'_y + a_{22}\beta'^2_y)}_{\tilde{a}_{22}} + 2i\underbrace{(a_{11}\alpha'_x\beta'_x + a_{12}(\alpha'_y\beta'_x + \alpha'_x\beta'_y) + \alpha'_y\beta'_y)}_{\tilde{a}_{12}} = 0 \end{aligned}$$

Отсюда по свойству комплексных чисел следует, что $\tilde{a}_{12} = 0$, $\tilde{a}_{11} = \tilde{a}_{22}$, так как $\tilde{a}_{11} - \tilde{a}_{22} = 0$. Тогда уравнение $\tilde{a}_{11}u_{\phi\phi}'' + 2\tilde{a}_{12}u_{\phi\psi}'' + \tilde{a}_{22}u_{\psi\psi}'' + \tilde{F} = 0$ превратится в следующее уравнение: $\tilde{a}_{11}u_{\alpha\alpha}'' + \tilde{a}_{22}u_{\beta\beta}'' = -\tilde{F}$ или $u_{\alpha\alpha}'' + u_{\beta\beta}'' = -\frac{\tilde{F}}{\tilde{a}_{11}} = \Phi(\alpha, \beta, u, u'_\alpha, u'_\beta)$ - канонический вид уравнения эллиптического типа.

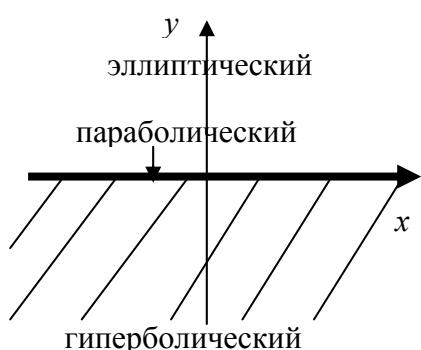
Итак: 1) $a_{12}^2 - a_{11}a_{22} > 0$ - гиперболический тип и $u_{xx}'' - u_{yy}'' = \Phi$ или $u_{xy}'' = \Phi$ - канонический вид уравнения;

2) $a_{12}^2 - a_{11}a_{22} < 0$ - эллиптический тип, а $u_{xx}'' + u_{yy}'' = \Phi$ - канонический вид уравнения;

3) $a_{12}^2 - a_{11}a_{22} = 0$ - параболический тип и $u_{xx}'' = \Phi$ - канонический вид уравнения.

Пример. Найти области гиперболичности, эллиптичности и параболичности уравнения $u_{xx}'' + yu_{yy}'' = 0$ и привести его к каноническому виду в области гиперболичности.

Дифференциальное уравнение в частных производных, содержащее явно все необходимые коэффициенты, имеет вид $a_{11}u_{xx}'' + 2a_{12}u_{xy}'' + a_{22}u_{yy}'' + F = 0$. В нашем случае $a_{11} = 1$, $a_{12} = 0$, $a_{22} = y$, то есть $a_{12}^2 - a_{11}a_{22} = 0 - 1 \cdot y = -y$. Тип уравнения зависит от знака выражения $a_{12}^2 - a_{11}a_{22}$. Если оно отрицательно, то это эллиптический тип, положительно - гиперболический, равно нулю - параболический:



- 1) $-y > 0$, $y < 0$ - область гиперболического типа;
- 2) $-y < 0$, $y > 0$ - область эллиптического типа;
- 3) $y = 0$ - область параболического типа.

Все области изображены на рисунке слева. Приведем теперь исходное уравнение к каноническому виду. Для этого решим уравнения (8.1.5):

$$\frac{dy}{dx} = \frac{a_{12} \pm \sqrt{a_{12}^2 - a_{11}a_{22}}}{a_{11}},$$

$$\frac{dy}{dx} = \frac{\pm \sqrt{-y}}{1}, \quad \pm \frac{dy}{\sqrt{-y}} = dx, \quad \mp \frac{d(-y)}{\sqrt{-y}} = dx,$$

$$\mp 2\sqrt{-y} = x + C. \text{ Итак, } \begin{cases} \varphi = -2\sqrt{-y} - x, & u_{\varphi\psi}'' = 0, \\ \psi = 2\sqrt{-y} - x. & \end{cases}$$

Пример. Привести к каноническому виду следующее дифференциальное уравнение $\frac{\partial^2 u}{\partial x^2} + 2 \sin x \frac{\partial^2 u}{\partial x \partial y} - \cos^2 x \frac{\partial^2 u}{\partial y^2} + \cos x \frac{\partial u}{\partial x} + \frac{1}{2} \sin 2x \frac{\partial u}{\partial y} = 0$.

Здесь $a_{11} = 1$, $a_{12} = \sin x$, $a_{22} = -\cos^2 x$. Составим и решим характеристическое уравнение: $\frac{dy}{dx} = \frac{a_{12} \pm \sqrt{a_{12}^2 - a_{11}a_{22}}}{a_{11}} = \frac{\sin x \pm \sqrt{\sin^2 x - 1 \cdot (-\cos^2 x)}}{1} = \sin x \pm 1$. Отсюда

$y = -\cos x \pm x + C$, то есть два первых интеграла характеристического уравнения имеют вид $\begin{cases} \varphi = C_1 = x - y - \cos x, \\ \psi = C_2 = x + y + \cos x. \end{cases}$ Обратное преобразование, нужное для нахождения производных

φ'_x , φ'_y , ψ'_x , ψ'_y , вычисления $\sin x$ и $\cos x$, также легко находится: $2x = \varphi + \psi \Rightarrow x = \frac{\varphi + \psi}{2}$,

$y = x - \cos x - \varphi = \frac{\varphi + \psi}{2} - \varphi - \cos x = \frac{\psi - \varphi}{2} - \cos x$. Так как $a_{12}^2 - a_{11}a_{22} = 1 > 0$, то исходное

уравнение – уравнение гиперболического типа. Его первая каноническая форма $u_{\varphi\psi}^{11} = -\frac{\tilde{F}}{2\tilde{a}_{12}}$,

где $\tilde{a}_{12} = a_{11}\varphi'_x\psi'_x + a_{12}(\varphi'_x\psi'_y + \varphi'_y\psi'_x) + a_{22}\varphi'_y\psi'_y$, $\tilde{F} = F(\varphi, \psi, u(\varphi, \psi), u'_\varphi, u'_\psi)$, а F определяется из исходного уравнения $a_{11}u''_{xx} + 2a_{12}u''_{xy} + a_{22}u''_{yy} + F(x, y, u(x, y), u'_x, u'_y) = 0$. В нашем случае

$F = \cos x \frac{\partial u}{\partial x} + \frac{1}{2} \sin 2x \frac{\partial u}{\partial y}$, $\cos x = \cos \frac{\varphi + \psi}{2}$, $\frac{1}{2} \sin 2x = \cos x \cdot \sin x = \cos \frac{\varphi + \psi}{2} \cdot \sin \frac{\varphi + \psi}{2}$. Далее

вычисляем необходимые производные: $\frac{\partial \varphi}{\partial x} = 1 + \sin x$, $\frac{\partial \varphi}{\partial y} = -1$, $\frac{\partial \psi}{\partial y} = 1 - \sin x$, $\frac{\partial \psi}{\partial y} = 1$,

$\frac{\partial u}{\partial x} = \frac{\partial u}{\partial \varphi} \cdot \frac{\partial \varphi}{\partial x} + \frac{\partial u}{\partial \psi} \cdot \frac{\partial \psi}{\partial x} = u'_\varphi(1 + \sin x) + u'_\psi(1 - \sin x)$, $\frac{\partial u}{\partial y} = \frac{\partial u}{\partial \varphi} \cdot \frac{\partial \varphi}{\partial y} + \frac{\partial u}{\partial \psi} \cdot \frac{\partial \psi}{\partial y} = u'_\varphi(-1) + u'_\psi$. Тогда

$$\tilde{F} = \cos \frac{\varphi + \psi}{2} [u'_\varphi(1 + \sin x) + u'_\psi(1 - \sin x)] + \sin \frac{\varphi + \psi}{2} \cos \frac{\varphi + \psi}{2} \left(\frac{\partial u}{\partial \varphi} - \frac{\partial u}{\partial \psi} \right) = u'_\varphi \cos \frac{\varphi + \psi}{2} +$$

$$+ u'_\varphi \sin \frac{\varphi + \psi}{2} \cos \frac{\varphi + \psi}{2} + u'_\psi \cos \frac{\varphi + \psi}{2} - u'_\psi \sin \frac{\varphi + \psi}{2} \cos \frac{\varphi + \psi}{2} + u'_\psi \cos \frac{\varphi + \psi}{2} \sin \frac{\varphi + \psi}{2} -$$

$$- u'_\varphi \cos \frac{\varphi + \psi}{2} \sin \frac{\varphi + \psi}{2} = \cos \frac{\varphi + \psi}{2} (u'_\varphi + u'_\psi),$$

$$\tilde{a}_{12} = 1 \cdot (1 + \sin x)(1 - \sin x) + \sin x [(1 \sin x) \cdot 1 + (-1) \cdot (1 - \sin x)] + (-\cos^2 x) \cdot (-1) = 1 - \sin^2 x + \\ + \sin x (1 + \sin x - 1 + \sin x) + \cos^2 x = 2 \cos^2 x + 2 \sin^2 x = 2,$$

$$-\frac{\tilde{F}}{2\tilde{a}_{12}} = -\frac{\cos \frac{\varphi + \psi}{2} (u'_\varphi + u'_\psi)}{2 \cdot 2} = -\frac{1}{4} \cos \frac{\varphi + \psi}{2} (u'_\varphi + u'_\psi). \text{ Тогда, окончательно, канонический вид}$$

исходного уравнения второго примера будет таким $u_{\varphi\psi}^{11} = -\frac{1}{4} (u'_\varphi + u'_\psi) \cos \frac{\varphi + \psi}{2}$.

8.2. Простейшие задачи, приводящие к дифференциальным уравнениям в частных производных

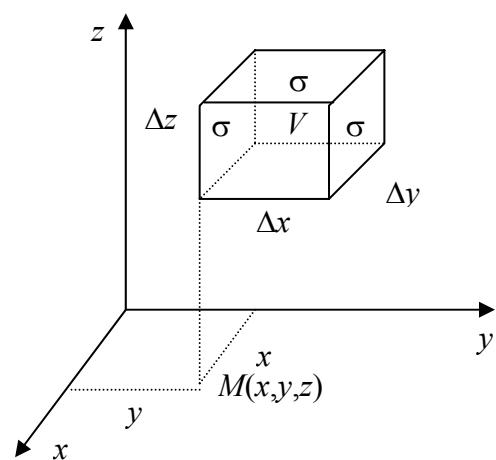
Температура тела (уравнение теплопроводности). Рассмотрим тело объемом V .

Пусть температура в его точке $M(x, y, z)$ в момент t равна $u = u(x, y, z, t)$. Покажем, что эта функция удовлетворяет уравнению

$$\frac{\partial u}{\partial t} = a^2 \left(\frac{\partial^2 u}{\partial x^2} + \frac{\partial^2 u}{\partial y^2} + \frac{\partial^2 u}{\partial z^2} \right) = a^2 \Delta u, \quad \text{где}$$

$\Delta = \frac{\partial^2}{\partial x^2} + \frac{\partial^2}{\partial y^2} + \frac{\partial^2}{\partial z^2}$ – оператор Лапласа или дельта-

оператор. Это уравнение называется уравнением теплопроводности. Рассмотрим левую и правую грани тела V . Количество тепла, проходящего через левую грань σ справа налево (из V) за промежуток $(t, t + \Delta t)$, с точностью до бесконечно малых



второго порядка равно: $\alpha \frac{\partial u(x, y, z)}{\partial x} \Delta y \Delta z \Delta t$. Действительно, α – коэффициент теплопровод-

ности тела, он постоянен везде в V , $\Delta y \Delta z$ - площадь σ , $\frac{\partial u}{\partial x}$ - скорость изменения температуры в направлении оси абсцисс. Эту скорость в пределах тела V также будем считать постоянной.

Количество тепла, проходящее через правую грань σ справа налево, также, очевидно, равно: $\alpha \frac{\partial u(x + \Delta x, y, z)}{\partial x} \Delta y \Delta z \Delta t$. Количество же тепла, оставшееся в V , за время Δt тогда

$$\begin{aligned} \text{равно: } & \alpha \frac{\partial u(x + \Delta x, y, z)}{\partial x} \Delta y \Delta z \Delta t - \alpha \frac{\partial u(x, y, z)}{\partial x} \Delta y \Delta z \Delta t = \alpha \left[\frac{\partial u(x + \Delta x, y, z)}{\partial x} - \frac{\partial u(x, y, z)}{\partial x} \right] \Delta y \Delta z \Delta t = \\ & = \alpha \left[\frac{\frac{\partial u(x + \Delta x, y, z)}{\partial x} - \frac{\partial u(x, y, z)}{\partial x}}{\Delta x} \cdot \Delta x \right] \Delta y \Delta z \Delta t \approx \alpha \frac{\partial^2 u}{\partial x^2} \Delta x \Delta y \Delta z \Delta t \end{aligned}$$

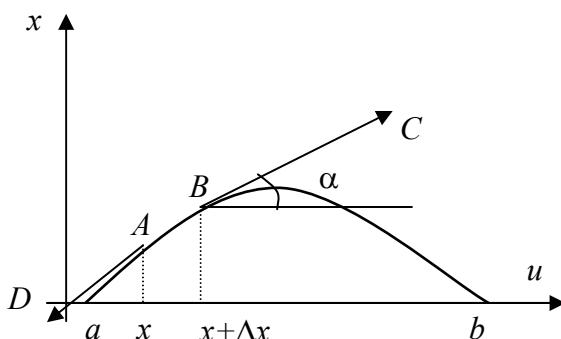
с точностью до членов второго порядка. Через две оставшиеся пары граней процесс передачи тепла будет идти так же. Следовательно, общее количество тепла, вошедшего в тело V через все грани, будет равно: $\alpha \left(\frac{\partial^2 u}{\partial x^2} + \frac{\partial^2 u}{\partial y^2} + \frac{\partial^2 u}{\partial z^2} \right) \Delta x \Delta y \Delta z \Delta t$.

Но, с другой стороны, это количество тепла равно: $\beta \frac{\partial u}{\partial t} \Delta x \Delta y \Delta z \Delta t$, где β - удельная теплоемкость тела, $\frac{\partial u}{\partial t}$ - скорость остывания или нагрева тела, $\Delta x \Delta y \Delta z = V$, Δt - время остывания или нагрева.

Тогда $\alpha \left(\frac{\partial^2 u}{\partial x^2} + \frac{\partial^2 u}{\partial y^2} + \frac{\partial^2 u}{\partial z^2} \right) \Delta x \Delta y \Delta z \Delta t = \beta \frac{\partial u}{\partial t} \Delta x \Delta y \Delta z \Delta t$ или $\alpha \Delta u = \beta \frac{\partial u}{\partial t}$. Окончательно, положив $a^2 = \frac{\alpha}{\beta}$, получим $\frac{\partial u}{\partial t} = a^2 \left(\frac{\partial^2 u}{\partial x^2} + \frac{\partial^2 u}{\partial y^2} + \frac{\partial^2 u}{\partial z^2} \right) = a^2 \Delta u$.

Данное уравнение имеет бесконечное множество решений. Чтобы их найти, надо наложить на функцию $u = u(x, y, z, t)$ дополнительные начальные и граничные условия.

Поперечные малые колебания струны. Струной называется тонкая нить, работающая на растяжение, но не на изгиб. Струна не сопротивляется изгибу. Пусть T - сила напряжения в струне постоянна в любой ее точке, ρ - линейная плотность массы. Концы струны закреплены в точках a и b . Рассмотрим функцию $u = u(x, t)$ - отклонение струны от оси абсцисс под действием силы. Будем считать отклонение $u = u(x, t)$ малым. На элемент Δx струны от точки A до точки B действуют две силы: натяжения $|\overline{BC}| = T$ и $|\overline{AD}| = T$. \overline{BC} приложена к точке B и направлена по касательной



к струне, причем $\tan \alpha = \frac{\partial u(x + \Delta x, t)}{\partial x}$.

$\sin \alpha \approx \frac{\partial u(x + \Delta x, t)}{\partial x}$. Проекция силы \overline{BC} на ось ou , очевидно, равна: $T \sin \alpha \approx T \frac{\partial u(x + \Delta x, t)}{\partial x}$.

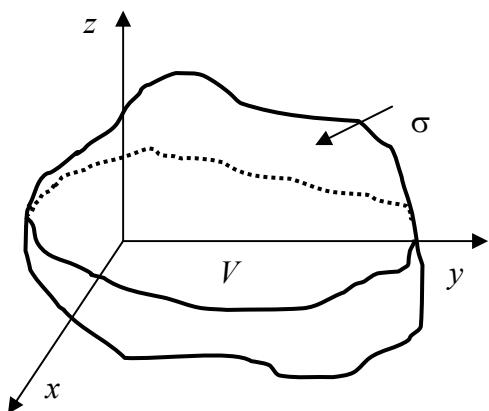
Проекция же силы \overline{AD} равна: $-T|_A \sin \alpha \approx -T \frac{\partial u(x, t)}{\partial x}$. Тогда сумма этих проекций будет

иметь следующий вид: $T \frac{\partial u(x + \Delta x, t)}{\partial x} - T \frac{\partial u(x, t)}{\partial x} = T \left[\frac{\frac{\partial u(x + \Delta x, t)}{\partial x} - \frac{\partial u(x, t)}{\partial x}}{\Delta x} \right] \Delta x \approx T \frac{\partial^2 u(x, t)}{\partial x^2} \Delta x$

с точностью до членов второго порядка.

С другой стороны, по второму закону Ньютона $\rho \Delta x \frac{\partial^2 u(x, t)}{\partial t^2}$ - сила, действующая на элемент Δx струны по направлению оси ou . Тогда $\rho \Delta x \frac{\partial^2 u(x, t)}{\partial t^2} = T \frac{\partial^2 u(x, t)}{\partial x^2} \Delta x$ или, если положить $a^2 = \frac{T}{\rho}$, то $\frac{\partial^2 u}{\partial t^2} = a^2 \frac{\partial^2 u}{\partial x^2}$ - уравнение поперечных колебаний струны в отсутствие внешней силы, то есть уравнение свободных колебаний. Если положить к тому же $y = at$, то уравнение превращается в $\frac{\partial^2 u}{\partial x^2} - \frac{\partial^2 u}{\partial y^2} = 0$.

Задачи, приводящие к уравнению Лапласа^{*}. Пусть имеется однородное тело V , ограниченное поверхностью σ . Температура в различных точках этого тела удовлетворяет уравнению теплопроводности: $\frac{\partial u}{\partial t} = a^2 \left(\frac{\partial^2 u}{\partial x^2} + \frac{\partial^2 u}{\partial y^2} + \frac{\partial^2 u}{\partial z^2} \right)$. Если процесс стационарный, то



$\frac{\partial u}{\partial t} = 0$ и данное уравнение превращается в $\frac{\partial^2 u}{\partial x^2} + \frac{\partial^2 u}{\partial y^2} + \frac{\partial^2 u}{\partial z^2} = \Delta u = 0$. Чтобы температура тела определялась однозначно, необходимо знать температуру на поверхности σ . Таким образом, при наличии источников тепла получим уравнение $\Delta u = -f(x, y, z)$ или $\Delta u = -\frac{F}{k}$, где F - плотность тепловых источников, k - коэффициент теплопроводности. Неоднородное уравнение $\Delta u = -f(x, y, z)$ называется уравнением Пуассона. Итак, задача о стационарном распределении температуры внутри тела V формулируется следующим образом: найти функцию $u(x, y, z)$, удовлетворяющую внутри V уравнению $\Delta u = -f(x, y, z)$ и специальному граничному условию, которое может быть взято в одном из следующих видов:

- 1) $u = f_1(x, y, z) \Big|_{\text{на } \sigma}$. Эта задача называется задачей Дирихле^{**} или первой краевой задачей;
- 2) $\frac{\partial u}{\partial n} = f_2(x, y, z) \Big|_{\text{на } \sigma}$, где \bar{n} - нормаль к поверхности σ , а $\frac{\partial u}{\partial n}$ - поток тепла через эту поверхность. Данная задача называется задачей Неймана или второй краевой задачей;
- 3) $\frac{\partial u}{\partial n} + h(u - f_3(x, y, z)) = 0 \Big|_{\text{на } \sigma}$ - третья краевая задача.

* Пьер Симон Лаплас (1749-1827) - французский математик.

** Петер Густав Лежен-Дирихле (1805-1859) - немецкий математик.

8.3. Граничные и начальные условия дифференциальных уравнений с частными производными*

Если уравнение (8.1.2) является уравнением параболического типа, то, как было показано в подразд. 8.1, с помощью замены независимых переменных, то есть путем перехода к некоторой криволинейной системе координат, это уравнение может быть преобразовано к виду $u_{xx}'' = \Phi$ или, подробно,

$$a(x,t) \frac{\partial^2 u}{\partial x^2} + d(x,t) \frac{\partial u}{\partial x} + e(x,t) \frac{\partial u}{\partial t} + g(x,t)u = f(x,t). \quad (8.3.1)$$

В физических задачах переменная x обычно играет роль пространственной координаты, а вторая независимая переменная – роль времени, поэтому для нее использовано обозначение t вместо y .

Уравнение (8.3.1) имеет, вообще говоря, бесчисленное множество решений, поэтому, чтобы обеспечить единственность решения и корректность задачи, необходимо, кроме самого уравнения, задать краевые и начальные условия аналогично тому, как это было сделано для обыкновенных дифференциальных уравнений в седьмой главе.

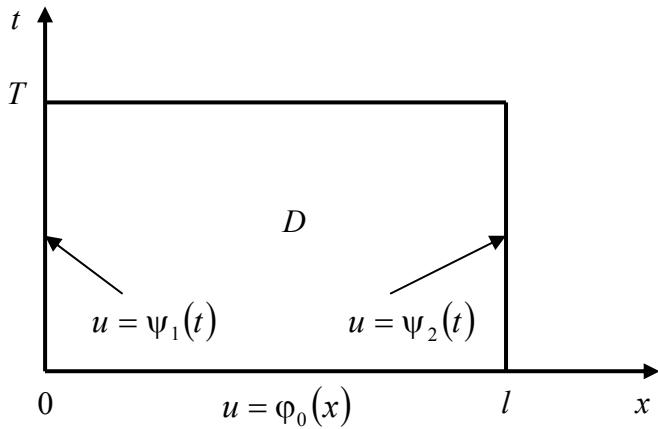
Рассмотрим важный для приложения в физике и технике случай, когда коэффициенты $a(x,t)$ и $e(x,t)$ имеют разные знаки, причем требуется найти решение уравнения (8.3.1) в прямоугольнике $D = \{(x,t) / 0 < x < l, 0 < t < T\}$. При этом должно быть выполнено начальное условие

$$u(x,0) = \varphi_0(x), \quad 0 \leq x \leq l, \quad (8.3.2)$$

где $\varphi_0(x)$ - заданная функция, определяющая значения решения $u(x,t)$ в начальный момент времени $t=0$. К условию (8.3.2) необходимо добавить краевые условия при $x=0$ и $x=l$, которые могут иметь, например, следующий вид:

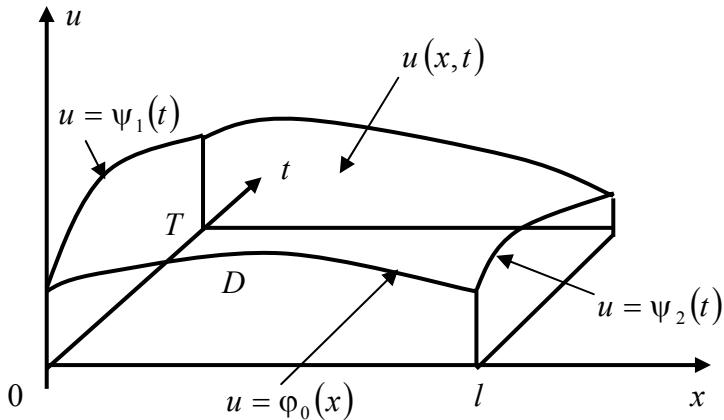
$$u(0,t) = \psi_1(t), \quad u(l,t) = \psi_2(t), \quad 0 \leq t \leq T, \quad (8.3.3)$$

где $\psi_1(t)$ и $\psi_2(t)$ - заданные функции. Если уравнение (8.3.1) описывает, например, изменение температуры в однородном стержне длины l (см. подразд. 8.4), то условия (8.3.3) означают, что температура в начальном и конечном сечениях стержня задана, в частности поддерживается постоянной. Геометрической интерпретацией искомого решения $u(x,t)$ является поверхность в пространстве (u,x,t) , которая проецируется в область D плоскости, причем согласно (8.3.2) – (8.3.3) заданы три кривые, которые представляют собой края этой поверхности и проецируются на нижнюю и боковые стороны прямоугольника D . Внутри прямоугольника и на его верхней стороне значения $u(x,t)$ неизвестны, поэтому форма поверхности неизвестна и ее требуется найти путем решения начально-краевой задачи (8.3.1)–(8.3.3) (см. рисунок).



и проецируются на нижнюю и боковые стороны прямоугольника D . Внутри прямоугольника и на его верхней стороне значения $u(x,t)$ неизвестны, поэтому форма поверхности неизвестна и ее требуется найти путем решения начально-краевой задачи (8.3.1)–(8.3.3) (см. рисунок).

* Подразд. 8.3 - 8.8 написаны проф. каф. Б4 А.Г. Кузьминым.



Если уравнение (8.1.2) является уравнением гиперболического типа, то, как показано в подразд. 8.1, с помощью некоторой замены независимых переменных это уравнение может быть приведено к виду $u_{xx}'' - u_{yy}'' = \Phi$ или

$$\frac{\partial^2 u}{\partial t^2} - a(x,t) \frac{\partial^2 u}{\partial x^2} + d(x,t) \frac{\partial u}{\partial x} + e(x,t) \frac{\partial u}{\partial t} + g(x,t)u = f(x,t), \quad (8.3.4)$$

где $a(x,t) > 0$. Для обеспечения единственности решения и корректности задачи необходимо задать начальные и краевые условия. Физически задачи, приводящие к уравнению (8.3.4), показывают, что обычно требуется решить уравнение (8.3.4) в прямоугольнике $D = \{(x,t) / 0 < x < l, 0 < t < T\}$ при начальных условиях

$$u(x,0) = \phi_0(x), \quad \left. \frac{\partial u}{\partial t} \right|_{t=0} = \phi_1(x), \quad 0 \leq x \leq l, \quad (8.3.5)$$

где $\phi_0(x)$, $\phi_1(x)$ - заданные функции. К условиям (8.3.5) следует добавить краевые условия,

например, вида (8.3.3). Геометрической интерпретацией искомого решения $u(x,t)$ по-прежнему является поверхность, которая проецируется на область D , причем согласно (8.3.3), (8.3.5) заданы три кривые, которые представляют собой края этой поверхности и проецируются на нижнюю и боковые стороны прямоугольника D . Внутри и на верхней стороне прямоугольника форма поверхности $u(x,t)$ неизвестна, и ее требуется найти путем решения поставленной начально-краевой задачи.

Уравнения параболического и

гиперболического типов (8.3.1) и (8.3.4) называют также нестационарными уравнениями математической физики, так как они описывают физические процессы, в которых пространственное распределение величины u изменяется с увеличением t , причем значения $u(x,t)$ зависят только от начальных и краевых условий при меньших значениях t . Как будет показано в подразд. 8.4 - 8.5, решения начально-краевых задач для нестационарных уравнений можно находить последовательно для возрастающих значений $t_0 = 0 < t_1 < t_2 < \dots < t_m = T$.

Если уравнение (8.1.2) является уравнением эллиптического типа, то с помощью замены независимых переменных оно может быть приведено к виду

$$u_{xx}'' + u_{yy}'' = \Phi \text{ или}$$

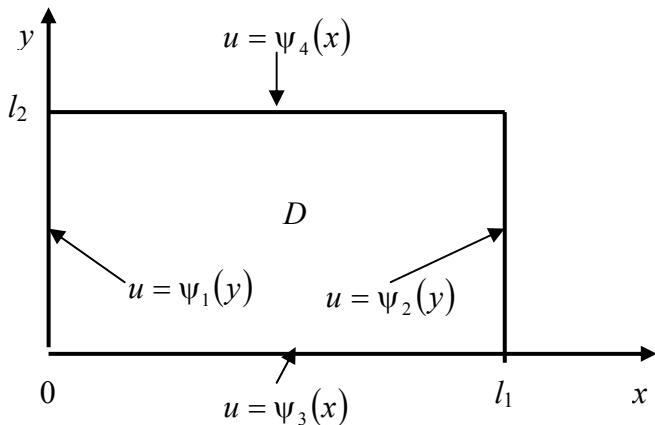
$$\frac{\partial^2 u}{\partial x^2} + \frac{\partial^2 u}{\partial y^2} + d(x,y) \frac{\partial u}{\partial x} + e(x,y) \frac{\partial u}{\partial y} + g(x,y)u = f(x,y). \quad (8.3.6)$$

Здесь оставлено обозначение (x, y) для независимых переменных, так как уравнения вида (8.3.6) описывают явления, в которых обе независимые переменные равноправны и играют роль пространственных координат. Уравнения эллиптического типа называются стационарными уравнениями математической физики.

Чтобы обеспечить единственность решения и корректность задачи, необходимо задать дополнительные условия. Физические соображения, приводящие к уравнению (8.3.6), показывают, что нужно задать краевые условия по всему периметру прямоугольника $D = \{(x, y) / 0 < x < l_1, 0 < y < l_2\}$, например, условия следующего вида:

$$u(0, y) = \psi_1(y), \quad u(l_1, y) = \psi_2(y), \quad 0 \leq y \leq l_2, \quad (8.3.7)$$

$$u(x, 0) = \psi_3(x), \quad u(x, l_2) = \psi_4(x), \quad 0 \leq x \leq l_1, \quad (8.3.8)$$



где $\psi_1(y), \psi_2(y), \psi_3(x), \psi_4(x)$ - заданные функции. В данном случае по всему периметру прямоугольника D заданы кривые, на которые опирается поверхность $u(x, t)$, а форму поверхности внутри прямоугольника необходимо найти путем решения краевой задачи (8.3.6) – (8.3.8).

8.4. Уравнения параболического типа. Явные и неявные схемы

В качестве примера рассмотрим уравнение

$$\frac{\partial u}{\partial t} - a_0^2 \frac{\partial^2 u}{\partial x^2} = f(x, t), \quad (8.4.1)$$

которое является частным случаем уравнения (8.3.1) и называется одномерным уравнением теплопроводности (см. подразд. 8.2). Коэффициент перед второй производной является здесь постоянной величиной; обозначение a_0^2 подчеркивает положительность этой величины. Из физики известно, что уравнение (8.4.1) описывает распространение тепла в однородном стержне постоянного сечения, при этом $u(x, t)$ - распределение температуры вдоль стержня в текущий момент времени t , x - продольная координата, $a_0^2 = \frac{\alpha}{\beta}$, α - коэффициент теплопроводности, β - удельная теплоемкость материала стержня.

Требуется найти решение уравнения (8.4.1) при начальных условиях (8.3.2) и краевых условиях (8.3.3).

Теорема 8.1. Задача (8.4.1), (8.3.2), (8.3.3) имеет единственное решение, которое непрерывно и дважды непрерывно дифференцируемо в замкнутой области D , если заданные функции $\phi_0(x), \psi_1(t), \psi_2(t), f(x, t)$ являются достаточно гладкими и, кроме того, выполнены условия согласования начального и краевых условий: $\phi_0(0) = \psi_1(0)$, $\phi_0(l) = \psi_2(0)$, $\psi'_1(0) = a_0^2 \phi''_0(0) + f(0, 0)$, $\psi'_2(0) = a_0^2 \phi''_0(l) + f(l, 0)$.

Доказательство

Необходимость выполнения двух последних условий следует из уравнения (8.4.1), записанного в точках $x = 0, t = 0$ и $x = l, t = 0$.

Докажем единственность решения методом от противного. Предположим, что существуют два решения $\bar{u}(x,t)$ и $\tilde{u}(x,t)$ поставленной задачи. Тогда их разность $u = \bar{u}(x,t) - \tilde{u}(x,t)$ удовлетворяет уравнению (8.4.1) с нулевой правой частью:

$$\frac{\partial u}{\partial t} - a_0^2 \frac{\partial^2 u}{\partial x^2} = \frac{\partial(\bar{u} - \tilde{u})}{\partial t} - a_0^2 \frac{\partial^2(\bar{u} - \tilde{u})}{\partial x^2} = f(x,t) - f(x,t) = 0, \quad (8.4.2)$$

а также нулевым краевым и начальному условиям

$$u(x,0) = 0, \quad 0 < x < l, \quad (8.4.3)$$

$$u(0,t) = 0, \quad u(l,t) = 0, \quad 0 < t < T, \quad (8.4.4)$$

которые следуют из условий (8.3.2), (8.3.3). Покажем, что решение $u(x,t)$ такой задачи тождественно равно нулю в области D . Умножим обе части уравнения (8.4.2) на $u(x,t)$ и проинтегрируем результат по D :

$$\iint_D \left(u \frac{\partial u}{\partial t} - a_0^2 u \frac{\partial^2 u}{\partial x^2} \right) dx dt = 0.$$

Левую часть полученного равенства преобразуем так:

$$\iint_D \left[\frac{1}{2} \frac{\partial(u^2)}{\partial t} - a_0^2 \frac{\partial}{\partial x} \left(u \frac{\partial u}{\partial x} \right) + a_0^2 \left(\frac{\partial u}{\partial x} \right)^2 \right] dx dt = 0. \quad (8.4.5)$$

Теперь разобьем интеграл в левой части на три интеграла и преобразуем первые два в интегралы по периметру прямоугольника с помощью формулы Грина*

$$\oint_{\Gamma} P dx + Q dt = \iint_D \left(\frac{\partial Q}{\partial x} - \frac{\partial P}{\partial t} \right) dx dt,$$

где Γ - граница области D , и интегрирование вдоль Γ происходит в направлении против часовой стрелки. Таким образом, из (8.4.5) получаем

$$-\oint_{\Gamma} \left(\frac{1}{2} u^2 dx + a_0^2 u \frac{\partial u}{\partial x} dt \right) + a_0^2 \iint_D \left(\frac{\partial u}{\partial x} \right)^2 dx dt = 0.$$

В силу условий (8.4.3) и (8.4.4) функция u равна нулю на боковых и на нижней сторонах прямоугольника D , поэтому от интеграла по контуру Γ в левой части последнего равенства остается только интеграл по верхней стороне прямоугольника:

$$-\int_l^0 \frac{1}{2} u^2(x,T) dx + a_0^2 \iint_D \left(\frac{\partial u}{\partial x} \right)^2 dx dt = 0.$$

Изменяя направление интегрирования, получаем

$$\frac{1}{2} \int_0^l u^2(x,T) dx + a_0^2 \iint_D \left(\frac{\partial u}{\partial x} \right)^2 dx dt = 0.$$

Под знаками обоих интегралов стоят неотрицательные функции, следовательно, это равенство возможно, только если каждый интеграл равен нулю. Таким образом,

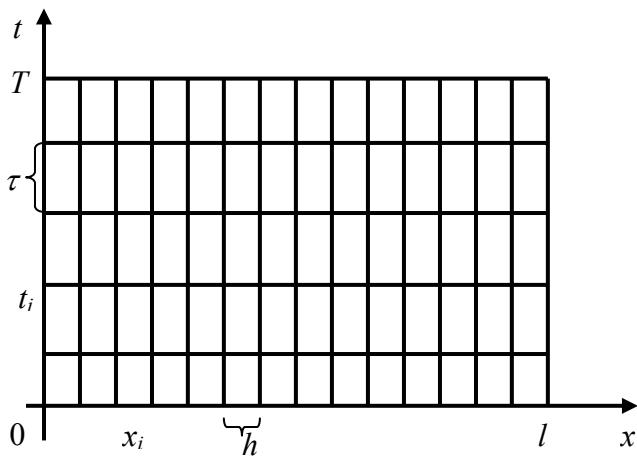
$$\iint_D \left(\frac{\partial u}{\partial x} \right)^2 dx dt = 0.$$

Это означает, что $\frac{\partial u}{\partial x} \equiv 0$ в D . Следовательно, $u(x,t)$ постоянна вдоль линий $t = \text{const}$, и так как на боковых сторонах $u(x,0) = u(x,l) = 0$, то $u(x,t) \equiv 0$ в D . Вспоминая определение u , находим $\bar{u}(x,t) \equiv \tilde{u}(x,t)$, что и требовалось доказать.

* Георг Грин (1793-1841) - английский математик.

Для нахождения приближенных значений решения задачи (8.4.1), (8.3.2), (8.3.3) в прямоугольнике D используем метод конечных разностей (метод сеток).

Разобьем отрезок $0 \leq x \leq l$ оси x на n равных частей длины $h = \frac{l}{n}$, при этом получим $n+1$ узловых точек $x_i = h \cdot i$, где $i = 0, 1, \dots, n$. Аналогично разобьем отрезок $0 \leq t \leq T$ оси t на m частей длины $\tau = \frac{T}{m}$, таким образом, получим $m+1$ узловых точек $t_j = \tau \cdot j$, где $j = 0, 1, \dots, m$. Проводя через все узловые точки прямые, параллельные осям координат, построим сетку, разбивающую область D на мелкие прямоугольные ячейки с вершинами (x_i, t_j) . Точное решение задачи, существование и единственность которого гарантированы



алгебраических уравнений, решение которой представляет собой приближенное решение $u_{i,j}$ поставленной задачи.

Итак, для аппроксимации производных от точного решения в узлах (x_i, t_j) $i = 1, 2, \dots, n-1$, $j = 1, 2, \dots, m-1$ области D используем формулы (4.1.1), (4.1.2) и (4.2.1):

$$\left. \frac{\partial^2 u}{\partial x^2} \right|_{i,j} = \frac{u(x_{i+1}, t_j) - 2u(x_i, t_j) + u(x_{i-1}, t_j)}{h^2} + O(h^2), \quad (8.4.6)$$

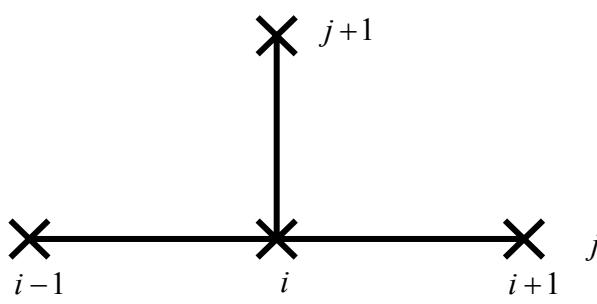
$$\left. \frac{\partial u}{\partial t} \right|_{i,j} = \frac{u(x_i, t_{j+1}) - u(x_i, t_j)}{\tau} + O(\tau). \quad (8.4.7)$$

Подставляя (8.4.6) и (8.4.7) в уравнение (8.4.1), находим

$$\frac{u(x_i, t_{j+1}) - u(x_i, t_j)}{\tau} - a_0^2 \frac{u(x_{i+1}, t_j) - 2u(x_i, t_j) + u(x_{i-1}, t_j)}{h^2} + O(\tau + h^2) = f(x_i, t_j). \quad (8.4.8)$$

Отбрасывая неизвестную погрешность $O(\tau + h^2)$, получаем приближенное алгебраическое уравнение

$$\frac{u_{i,j+1} - u_{i,j}}{\tau} - a_0^2 \frac{u_{i+1,j} - 2u_{i,j} + u_{i-1,j}}{h^2} = f(x_i, t_j), \quad (8.4.9)$$



которое связывает значения решения в четырех соседних узлах, расположенных так, как показано на рисунке, и которое справедливо при $i = 1, 2, \dots, n-1$, $j = 0, 1, \dots, m-1$.

Начальные и краевые условия (8.3.2) и (8.3.3) определяют следующие величины:

$$u_{i,0} = \varphi_0(x_i), \quad i = 0, 1, \dots, n. \quad (8.4.10)$$

$$u_{0,j} = \psi_1(t_j), \quad u_{n,j} = \psi_2(t_j), \quad j = 0, 1, \dots, m. \quad (8.4.11)$$

Поэтому система (8.4.9) содержит $m(n-1)$ уравнений и столько же неизвестных, которыми являются значения приближенного решения во внутренних узлах области D и в $n-1$ узлах, расположенных на верхней стороне прямоугольника.

Уравнение (8.4.9) называется явной разностной схемой для уравнения теплопроводности, так как оно позволяет найти решение последовательно на всех слоях по времени $t = t_j$, $j = 1, 2, \dots, m$. Действительно, при $j = 0$ из (8.4.9) получаем

$$\frac{u_{i,1} - u_{i,0}}{\tau} - a_0^2 \frac{u_{i+1,0} - 2u_{i,0} + u_{i-1,0}}{h^2} = f(x_i, t_0), \quad i = 1, 2, \dots, n-1. \quad (8.4.12)$$

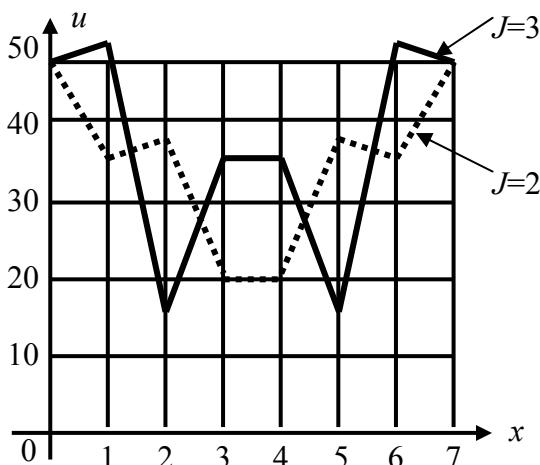
В этом равенстве значения $u_{i,0}$, $u_{i-1,0}$, $u_{i+1,0}$ известны в силу (8.4.10), поэтому легко найти $u_{i,1}$ при $i = 1, 2, \dots, n-1$. Значения $u_{i,1}$ в крайних точках $i=0$ и $i=n$ находятся из краевых условий (8.4.11). Таким образом, значения приближенного решения оказываются известными на слое $t = t_1$. Теперь, полагая $j = 1$, из (8.4.9) можно аналогично найти $u_{i,2}$ на следующем и на других слоях и вычислить приближенное решение во всех узлах сетки вплоть до узлов, расположенных на верхней стороне прямоугольника D .

Свойства полученного приближенного решения существенно зависят от того, выполнено ли соотношение

$$\frac{\tau a_0^2}{h^2} \leq \frac{1}{2} \quad (8.4.13)$$

между шагами h и τ вычислительной сетки или нет. Если условие (8.4.13) не выполнено, то полученное приближенное решение $u_{i,j}$ имеет колебательный характер, причем амплитуда колебаний быстро нарастает с ростом t , что не имеет ничего общего с точным решением исходной задачи. Для примера представлены графики приближенного решения после второго и

третьего шага по времени с $\tau = 0.9$, $h = 1$ и $a_0 = 1$, $f \equiv 0$, при этом использовались начальные данные $u(x_i, 0) = 50, 30, 20, 10, 10, 20, 30, 50$ и краевые условия $u(0, t_j) = u(7, t_j) = 50$. Такой колебательный характер приближенного решения связан со свойствами уравнения (8.4.9), определяющего $u_{i,j}$. Если же условие (8.4.13) выполнено, то приближенное решение оказывается достаточно гладким, и при малых шагах сетки оно является хорошим приближением к точному решению. Как можно доказать, при выполнении условия устойчивости (8.4.13) справедлива следующая оценка для глобальной по-



грешности полученного решения:

$$\max_{i,j} |u(x_i, t_j) - u_{i,j}| = O(\tau + h^2). \quad (8.4.14)$$

В частном случае, когда в условии (8.4.13) выбран знак равенства $2\tau a_0^2 = h^2$, формула (8.4.9) принимает особенно простой вид:

$$u_{i,j+1} - u_{i,j} - \frac{u_{i+1,j} - 2u_{i,j} + u_{i-1,j}}{2} = \tau f(x_i, t_j), \text{ то есть}$$

$$u_{i,j+1} = \frac{u_{i+1,j} + u_{i-1,j}}{2} + \tau f(x_i, t_j).$$

Отсюда видно свойство сглаживания значений решения уравнения теплопроводности: если в точке стержня с координатой x_i нет источников тепла, так что $f(x_i, t_j) = 0$, то значение ре-

шения в точке x_i на следующем шаге по времени равно полусумме его значений в соседних узлах x_{i+1} и x_{i-1} в данный момент времени.

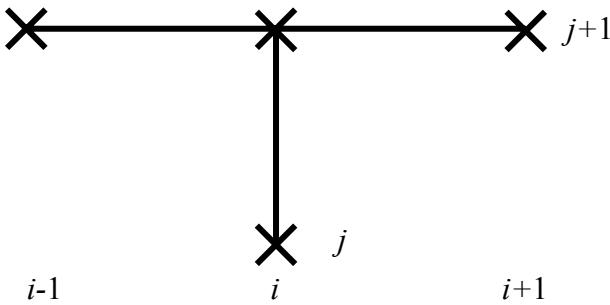
Условие устойчивости (8.4.13) накладывает ограничение на выбор шага τ по времени. Если пространственный шаг h имеет, например, порядок 10^{-3} , а коэффициент a_0 - порядок единицы, то τ ограничено величиной порядка 10^{-6} , следовательно, требуется очень большое число шагов по времени t , чтобы получить решение во всей расчетной области D .

Чтобы избежать этой трудности, можно использовать другой вычислительный алгоритм, который называется неявной разностной схемой для уравнения теплопроводности и строится следующим образом.

Запишем разностные аппроксимации производных, входящих в уравнение (8.4.1):

$$\frac{\partial^2 u}{\partial x^2} \Big|_{i,j+1} = \frac{u(x_{i+1}, t_{j+1}) - 2u(x_i, t_{j+1}) + u(x_{i-1}, t_{j+1})}{h^2} + O(h^2), \quad (8.4.15)$$

$$\frac{\partial u}{\partial t} \Big|_{i,j+1} = \frac{u(x_i, t_{j+1}) - u(x_i, t_j)}{\tau} + O(\tau). \quad (8.4.16)$$



При этом используется шаблон, представленный на следующем рисунке. Подставляя (8.4.15) и (8.4.16) в уравнение (8.4.1), находим соотношение

$$\frac{u(x_i, t_{j+1}) - u(x_i, t_j)}{\tau} - a_0^2 \frac{u(x_{i+1}, t_{j+1}) - 2u(x_i, t_{j+1}) + u(x_{i-1}, t_{j+1})}{h^2} + O(\tau + h^2) = f(x_i, t_{j+1}).$$

Отбрасывая погрешность $O(\tau + h^2)$, получаем приближенное уравнение

$$\frac{u_{i,j+1} - u_{i,j}}{\tau} - a_0^2 \frac{u_{i+1,j+1} - 2u_{i,j+1} + u_{i-1,j+1}}{h^2} = f(x_i, t_{j+1}), \quad (8.4.17)$$

справедливое при $i = 1, 2, \dots, n-1$, $j = 0, 1, \dots, m-1$, которое связывает значения решения в четырех соседних узлах, показанных на предыдущем рисунке.

К уравнению (8.4.17) нужно присоединить равенства (8.4.10) и (8.4.11), которые следуют из начальных и краевых условий. Полученную систему алгебраических уравнений можно решить, последовательно определяя значения приближенного решения на слоях $t = t_j$, $j = 1, 2, \dots, m$. Однако в отличие от системы (8.4.9) значения решения на последующем слое не вычисляются по явной формуле, а должны быть найдены из системы $n-1$ уравнений. Действительно, при $j = 0$ соотношение (8.4.17) принимает вид

$$\frac{u_{i,1} - u_{i,0}}{\tau} - a_0^2 \frac{u_{i+1,1} - 2u_{i,1} + u_{i-1,1}}{h^2} = f(x_i, t_1). \text{ Здесь } u_{i,0} \text{ известно в силу условия (8.4.10), поэтому получаем трехдиагональную систему уравнений относительно неизвестных } u_{i,1},$$

которая может быть решена методом прогонки, описанным в подразд. 5.4. После нахождения приближенного решения на первом слое по времени можно аналогично с помощью прогонки вычислить значения решения на втором слое и так далее.

Применение неявной разностной схемы (8.4.17) не требует выполнения какого-либо соотношения между шагами h и τ , которое обеспечивало бы устойчивость решения алгебраической системы уравнений и отсутствие нефизических колебаний. Поэтому вычисления

по схеме (8.4.17) можно проводить с использованием большего значения τ , чем при расчетах по явной схеме. Однако глобальная погрешность полученного решения по-прежнему определяется величиной $O(\tau + h^2)$, которая содержит τ в первой степени и поэтому не позволяет получать решение высокой точности при больших шагах по времени.

В связи с этим возникает необходимость повышения точности аппроксимации производной $\frac{\partial u}{\partial t}$, входящей в дифференциальное уравнение (8.4.1), и использования схемы, аналогичной той, которая представлена модифицированным методом Эйлера для обыкновенного дифференциального уравнения первого порядка (см. подразд. 7.4). Как можно показать, для значений производных в промежуточном узле $\left(x_i, t_{\frac{j+1}{2}}\right)$, где $t_{\frac{j+1}{2}} = t_j + \frac{\tau}{2}$, справедливы

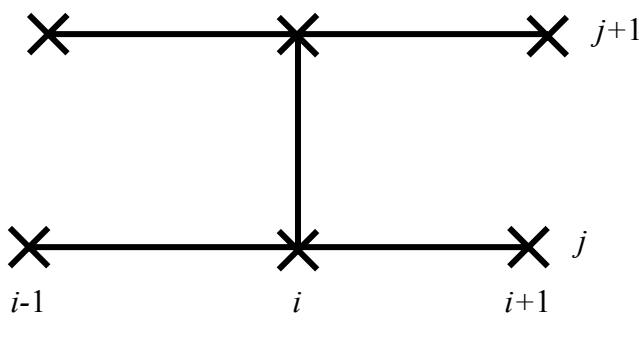
следующие приближения:

$$\frac{\partial^2 u}{\partial x^2} \Big|_{i, \frac{j+1}{2}} = \frac{u(x_{i+1}, t_j) - 2u(x_i, t_j) + u(x_{i-1}, t_j) + u(x_{i+1}, t_{j+1}) - 2u(x_i, t_{j+1}) + u(x_{i-1}, t_{j+1})}{2h^2} + O(h^2), \quad (8.4.18)$$

$$\frac{\partial u}{\partial t} \Big|_{i, \frac{j+1}{2}} = \frac{u(x_i, t_{j+1}) - u(x_i, t_j)}{\tau} + O(\tau^2), \quad (8.4.19)$$

$i = 1, 2, \dots, n-1$, $j = 0, 1, \dots, m-1$. Здесь использованы значения решения в трех точках, расположенных на слое t_j , и трех соседних точках, расположенных на слое t_{j+1} (см. следующий рисунок). Схема, полученная с помощью (8.4.18) и (8.4.19), называется шеститочечной неявной

схемой Кранка - Николсон*. Подставляя (8.4.18) и (8.4.19) в уравнение (8.4.1) и отбрасывая погрешности, получаем систему алгебраических уравнений, которую можно решать по слоям методом прогонки так же, как и систему (8.4.17). В схеме Кранка - Николсон глобальная оценка для погрешности полученного решения имеет вид $\max_{i,j} |u(x_i, t_j) - u_{i,j}| = O(\tau^2 + h^2)$.



В заключение отметим некоторые свойства одномерного уравнения переноса

$$\frac{\partial u}{\partial t} - a_0^2 \frac{\partial^2 u}{\partial x^2} + \alpha(x, t) \frac{\partial u}{\partial x} = f(x, t), \quad (8.4.20)$$

которое отличается от (8.4.1) наличием слагаемого с первой производной по пространственной координате. С физической точки зрения появление слагаемого $\alpha \frac{\partial u}{\partial x}$ в уравнении означает учет конвекции (переноса) тепла за счет движения теплопроводящей среды вдоль оси абсцисс. Например, уравнение (8.4.20) описывает распределение температуры в потоке жидкости, движущейся вдоль оси абсцисс со скоростью $\alpha(x, t)$.

Как и уравнение теплопроводности, уравнение переноса имеет параболический тип, поэтому начальное и краевые условия для него записываются в том же виде: (8.3.2), (8.3.3). Однако при больших значениях отношения α/a_0^2 основную роль играют два первые члена, и это уравнение проявляет свойства, аналогичные свойствам уравнения чистой конвекции:

* Джон Кранк (р.1916) - английский математик, Филлип Николсон (1917-1968) – английский математик.

$$\frac{\partial u}{\partial t} + \alpha(x, t) \frac{\partial u}{\partial x} = f(x, t).$$

Постановка корректных задач для последнего уравнения принципиально отличается от постановки задачи (8.4.20), (8.3.2), (8.3.3). Например, при $\alpha(x, t) > 0$, кроме начального условия (8.3.2), необходимо задать краевые условия только на левой стороне прямоугольника D , а правая сторона должна быть свободна от каких-либо условий на искомое решение $u(x, t)$. В связи с этим могут возникать существенные трудности при численном решении уравнения переноса с большими значениями α/a_0^2 . В частности, аппроксимация первой производной решения по x центральным разностным отношением

$$\left. \frac{\partial u}{\partial x} \right|_{i,j} = \frac{u_{i+1,j} - u_{i-1,j}}{2h} + O(h^2)$$

приводит к алгоритму, который оказывается существенно неустойчивым. В то же время различные варианты аппроксимации этой производной односторонними разностными отношениями, сдвинутыми «вверх по потоку», обеспечивают устойчивость схемы.

8.5. Уравнения гиперболического типа

В качестве примера уравнения гиперболического типа рассмотрим уравнение

$$\frac{\partial^2 u}{\partial t^2} - a_0^2 \frac{\partial^2 u}{\partial x^2} = f(x, t), \quad (8.5.1)$$

которое описывает поперечные колебания упругой натянутой струны, где x является продольной координатой, t - временем, а $u(x, t)$ - смещением участков струны в поперечном направлении. Коэффициент a_0 определяется материалом струны и силой ее натяжения:

$$a_0 = \frac{T}{\rho}, \text{ где } T \text{ - сила натяжения, } \text{Н/м}^2, \text{ а } \rho \text{ - плотность материала, } \text{кг/м}.$$

Требуется найти решение уравнения колебаний струны (8.5.1) в прямоугольнике $D = \{(x, t) / 0 < x < l, 0 < t < T\}$ при начальных условиях (8.3.5) и краевых условиях (8.3.3).

Начальные условия означают, что задано смещение и скорость участков струны в начальный момент времени. Краевые условия означают, что задано смещение начала и конца струны при всех $t \geq 0$; в частности, если $u(0, t) = u(l, t) = 0$, то концы струны являются жестко закрепленными.

Теорема 8.2. Задача (8.5.1), (8.3.3), (8.3.5) имеет единственное решение, которое непрерывно и дважды непрерывно дифференцируемо в замкнутой области D , если заданные функции $\varphi_0(x), \varphi_1(x), \psi_1(t), \psi_2(t), f(x, t)$ являются достаточно гладкими и, кроме того, выполнены условия согласования начальных и краевых данных:

$$\begin{aligned} \psi_1(0) &= \varphi_0(0), \quad \psi_2(0) = \varphi_0(l), \quad \psi'_1(0) = \varphi_1(0), \quad \psi'_2(l) = \varphi_1(l), \\ \psi''_1(0) &= a_0^2 \varphi''_0(0) + f(0, 0), \quad \psi''_2(0) = a_0^2 \varphi''_0(l) + f(l, 0). \end{aligned}$$

Последние два условия следуют из уравнения (8.5.1) при $x = 0, t = 0$ и $x = l, t = 0$. Для нахождения приближенных значений решения задачи (8.5.1), (8.3.3), (8.3.5) используем метод конечных разностей. Как и в подразд. 8.4, разобьем отрезок $0 \leq x \leq l$ оси абсцисс на n частей длины: $h = l/n$, отрезок $0 \leq t \leq T$ оси t - на m частей длины $\tau = T/m$ и построим в области D прямоугольную равномерную сетку (см. рисунок на с. 200). В узлах этой сетки точное решение задачи, существование и единственность которого доказана теоремой 8.2, принимает значения $u(x_i, t_j)$.

Получим систему алгебраических уравнений относительно $u(x_i, t_j)$. Для аппрокси-

мации производных от решения в узлах (x_i, t_j) , $i = 1, 2, \dots, n-1$, $j = 1, 2, \dots, m-1$ используем центральные разностные отношения:

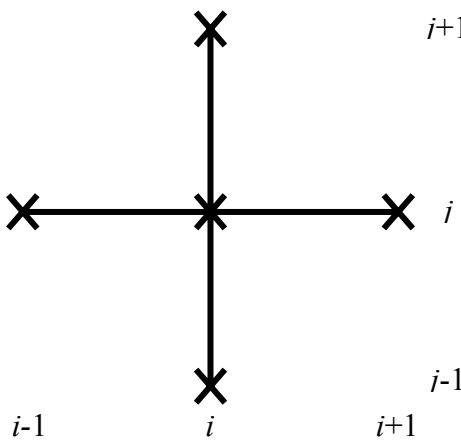
$$\frac{\partial^2 u}{\partial x^2} \Big|_{i,j} = \frac{u(x_{i+1}, t_j) - 2u(x_i, t_j) + u(x_{i-1}, t_j)}{h^2} + O(h^2), \quad (8.5.2)$$

$$\frac{\partial^2 u}{\partial t^2} \Big|_{i,j} = \frac{u(x_i, t_{j+1}) - 2u(x_i, t_j) + u(x_i, t_{j-1})}{\tau^2} + O(\tau^2). \quad (8.5.3)$$

Соответствующий пятиточечный разностный шаблон показан на рисунке.

Подставляя (8.5.2) и (8.5.3) в уравнение (8.5.1), находим

$$\frac{u(x_i, t_{j+1}) - 2u(x_i, t_j) + u(x_i, t_{j-1})}{\tau^2} - a_0^2 \frac{u(x_{i+1}, t_j) - 2u(x_i, t_j) + u(x_{i-1}, t_j)}{h^2} + O(\tau^2 + h^2) = f(x_i, t_j). \quad (8.5.4)$$



Отбрасывая неизвестную погрешность $O(\tau^2 + h^2)$, получаем приближенное уравнение

$$\frac{u_{i,j+1} - 2u_{i,j} + u_{i,j-1}}{\tau^2} - a_0^2 \frac{u_{i+1,j} - 2u_{i,j} + u_{i-1,j}}{h^2} = f(x_i, t_j), \quad (8.5.5)$$

где $i = 1, 2, \dots, n-1$, $j = 0, 1, \dots, m-1$. К этому уравнению следует присоединить условия, которые следуют из краевых и начальных данных (8.3.3) и (8.3.5):

$$u_{0,j} = \psi_1(t_j), \quad u_{n,j} = \psi_2(t_j), \quad j = 0, 1, \dots, m. \quad (8.5.6)$$

$$u_{i,0} = \phi_0(x_i), \quad u_{i,1} - u_{i,0} = \tau \phi_1(x_i), \quad i = 0, 1, \dots, n. \quad (8.5.7)$$

Уравнение (8.5.5) представляет собой явную разностную схему для уравнения колебаний струны, так как с учетом (8.5.6) и (8.5.7) оно позволяет найти решение последовательно на всех слоях по времени $t = t_j$, $j = 1, 2, \dots, m$. Действительно, на первом слое $j = 1$ значения $u_{i,1}$ легко вычисляются с помощью (8.5.7). Далее перепишем (8.5.5), оставляя в левой части только значение искомого решения на слое $j + 1$:

$$u_{i,j+1} = 2u_{i,j} - u_{i,j-1} - \frac{\tau^2 a_0^2}{h^2} (u_{i+1,j} - 2u_{i,j} + u_{i-1,j}) + \tau^2 f(x_i, t_j). \quad (8.5.8)$$

При $j = 1$ получаем

$$u_{i,2} = 2u_{i,1} - u_{i,0} - \frac{\tau^2 a_0^2}{h^2} (u_{i+1,1} - 2u_{i,1} + u_{i-1,1}) + \tau^2 f(x_i, t_1), \quad i = 1, 2, \dots, n-1.$$

В этом равенстве $u_{i,1}$, $u_{i-1,1}$, $u_{i+1,1}$, $u_{i,0}$ известны в силу (8.5.7), поэтому легко найти $u_{i,2}$ при $i = 1, 2, \dots, n-1$. Значения $u_{i,2}$ в крайних точках $i = 0$ и $i = n$ определяются из краевых условий (8.5.6). Поэтому значения приближенного решения оказываются известными на всем слое $t = t_2$. Теперь, полагая $j = 2$, из (8.5.8) можно найти $u_{i,3}$ на следующем слое и так далее аналогично определить приближенное решение во всех узлах сетки вплоть до узлов, расположенных на верхней стороне прямоугольника D .

Чтобы увеличить точность нахождения приближенного решения на первом слое $t = t_1$, вместо формулы $u_{i,1} - u_{i,0} = \tau \phi_1(x_i)$, которая получена из аппроксимации начального условия $\phi_1(x_i) = \frac{\partial u}{\partial t} \Big|_{t=0} = \frac{u_{i,1} - u_{i,0}}{\tau} + O(\tau)$, можно использовать другое соотношение, которое следует из разложения точного решения по формуле Тейлора:

$$u(x_i, t_1) = u(x_i, 0) + \tau \frac{\partial u}{\partial t} \Big|_{x=x_i, t=0} + \frac{\tau^2}{2} \frac{\partial^2 u}{\partial t^2} \Big|_{x=x_i, t=0} + O(\tau^3). \quad (8.5.9)$$

В этой формуле первая производная равна $\varphi_1(x_i)$. Для нахождения второй производной используем уравнение (8.5.1):

$$\left. \frac{\partial^2 u}{\partial t^2} \right|_{x=x_i, t=0} = a_0^2 \left. \frac{\partial^2 u}{\partial x^2} \right|_{x=x_i, t=0} + f(x_i, 0) = a_0^2 \varphi_0''(x_i) + f(x_i, 0).$$

Подставляя это выражение в (8.5.9) и опуская погрешность аппроксимации, получаем окончательно $u_{i,1} = u_{i,0} + \tau \varphi_1(x_i) + \frac{\tau^2}{2} [a_0^2 \varphi_0''(x_i) + f(x_i, 0)]$.

Свойства полученного приближенного решения $u_{i,j}$ существенно зависят от того, выполнено или нет условие устойчивости

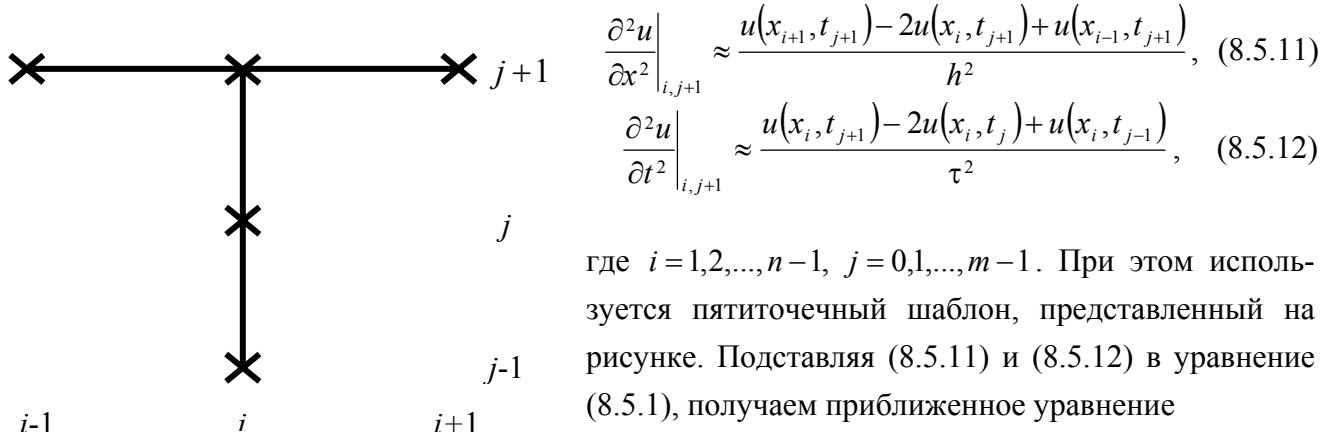
$$\frac{\tau a_0}{h} \leq 1. \quad (8.5.10)$$

Если выполнено, то при достаточно малых шагах сетки $u_{i,j}$ является хорошим приближением к точному решению и справедлива следующая оценка для глобальной погрешности:

$$\max_{0 \leq j \leq m} \left\{ \frac{1}{n+1} \sum_{i=0}^n [u(x_i, t_j) - u_{i,j}]^2 \right\}^{\frac{1}{2}} = O(h^2 + \tau^2).$$

Заметим, что, кроме колебаний струны, уравнение (8.5.1) описывает распространение плоских волн сжатия в упругой сплошной среде, например в атмосфере или слое жидкости. Параметр a_0 имеет смысл скорости распространения волн, которая может достигать очень больших значений (несколько километров в секунду в задачах гидроакустики). При этом условие устойчивости (8.5.10) накладывает сильное ограничение на выбор величины τ , что существенно увеличивает необходимое компьютерное время и затраты на проведение расчетов. Чтобы избежать ограничения на выбор τ , связанного с необходимостью выполнения условия (8.5.10), можно использовать неявную разностную схему, которая строится следующим образом.

Запишем разностные аппроксимации производных, входящих в уравнение (8.5.1), в узле (x_i, t_{j+1}) , используя центральное разностное отношение для $\frac{\partial u}{\partial x^2}$ и разностное отношение «со смещением» - для $\frac{\partial u}{\partial t^2}$:



$$\frac{u_{i,j+1} - 2u_{i,j} + u_{i,j-1}}{\tau^2} - a_0^2 \frac{u_{i+1,j+1} - 2u_{i,j+1} + u_{i-1,j+1}}{h^2} = f(x_i, t_{j+1}), \quad (8.5.13)$$

справедливое при $i = 1, 2, \dots, n-1$, $j = 0, 1, \dots, m-1$.

К уравнению (8.5.13) следует присоединить соотношения (8.5.6) и (8.5.7), которые следуют из начальных и краевых условий. Полученную систему алгебраических уравнений

можно решить, последовательно определяя значения приближенного решения на слоях $t = t_j$, $j = 2, 3, \dots, m$ методом прогонки. Действительно, при $j = 1$ соотношение (8.5.13) принимает вид

$$\frac{u_{i,2} - 2u_{i,1} + u_{i,0}}{\tau^2} - a_0^2 \frac{u_{i+1,2} - 2u_{i,2} + u_{i-1,2}}{h^2} = f(x_i, t_2),$$

где $u_{i,0}$, $u_{i,1}$ известны в силу условий (8.5.7), поэтому получаем трехдиагональную систему уравнений относительно неизвестных $u_{i,2}$, которая может быть решена методом, описанным в подразд. 5.4. Значения $u_{i,2}$ в крайних точках $i = 0$, $i = n$, как обычно, находятся из краевых условий (8.5.6). После нахождения приближенного решения на слое $t = t_2$ аналогично можно вычислить значения решения на слое $t = t_3$ и так далее. Применение неявной разностной схемы (8.5.13) не требует выполнения какого-либо соотношения между шагами h и τ , которое обеспечивало бы устойчивость решения алгебраической системы уравнений.

8.6. Уравнения эллиптического типа

В качестве примера рассмотрим уравнение

$$\frac{\partial^2 u}{\partial x^2} + \frac{\partial^2 u}{\partial y^2} = f(x, y), \quad (8.6.1)$$

которое является частным случаем уравнения (8.3.6). Если $f(x, y) \equiv 0$, то уравнение (8.6.1) называется уравнением Лапласа.

Задача нахождения решения уравнения (8.6.1) в области $D = \{(x, y) / 0 < x < l_1, 0 < y < l_2\}$ при краевых условиях (8.3.7) и (8.3.8), заданных на всей границе рассматриваемой области, называется задачей Дирихле.

Теорема 8.3. **Задача (8.6.1), (8.3.7), (8.3.8) имеет единственное решение, которое непрерывно и дважды непрерывно дифференцируемо в замкнутой области D , если заданные функции $\psi_1(y)$, $\psi_2(y)$, $\psi_3(x)$, $\psi_4(x)$, $f(x, y)$ являются достаточно гладкими и, кроме того, выполняются условия согласования:**

$$\begin{aligned} \psi_1(0) &= \psi_3(0), \quad \psi_1(l_2) = \psi_4(0), \quad \psi_2(0) = \psi_3(l_1), \quad \psi_2(l_2) = \psi_4(l_1), \\ \psi_1''(0) + \psi_3''(0) &= f(0, 0), \quad \psi_1''(l_2) + \psi_4''(0) = f(0, l_2), \\ \psi_2''(0) + \psi_3''(l_1) &= f(l_1, 0), \quad \psi_2''(l_2) + \psi_4''(l_1) = f(l_1, l_2). \end{aligned}$$

Докажем единственность решения. Предположим, что существуют два решения $\bar{u}(x, y)$ и $\tilde{u}(x, y)$ поставленной задачи. Тогда их разность $u = \bar{u}(x, y) - \tilde{u}(x, y)$ является решением уравнения (8.6.1) с нулевой правой частью:

$$\frac{\partial^2 u}{\partial x^2} + \frac{\partial^2 u}{\partial y^2} = \frac{\partial^2(\bar{u} - \tilde{u})}{\partial x^2} + \frac{\partial^2(\bar{u} - \tilde{u})}{\partial y^2} = f(x, y) - f(x, y) = 0 \quad (8.6.2)$$

и обращается в нуль на всем периметре прямоугольника D в силу заданных краевых условий

$$u(0, y) = 0, \quad u(l_1, y) = 0, \quad 0 \leq y \leq l_2, \quad (8.6.3)$$

$$u(x, 0) = 0, \quad u(x, l_2) = 0, \quad 0 \leq x \leq l_1. \quad (8.6.4)$$

Покажем, что решение $u(x, y)$ такой задачи тождественно равно нулю в области D . Умножим обе части уравнения (8.6.2) на $u(x, y)$ и проинтегрируем результат по области D :

$$\iint_D \left(u \frac{\partial^2 u}{\partial x^2} + u \frac{\partial^2 u}{\partial y^2} \right) dx dy = 0.$$

Левую часть полученного равенства преобразуем так:

$$\iint_D \left[\frac{\partial}{\partial x} \left(u \frac{\partial u}{\partial x} \right) - \left(\frac{\partial u}{\partial x} \right)^2 + \frac{\partial}{\partial y} \left(u \frac{\partial u}{\partial y} \right) - \left(\frac{\partial u}{\partial y} \right)^2 \right] dx dy = 0. \quad (8.6.5)$$

Теперь разобьем двойной интеграл на четыре интеграла и преобразуем первый и третий с помощью формулы Грина

$$\oint_{\Gamma} P dx + Q dy = \iint_D \left(\frac{\partial Q}{\partial x} - \frac{\partial P}{\partial y} \right) dx dy,$$

где Γ - граница области D и интегрирование вдоль контура Γ происходит в направлении против часовой стрелки. Тогда из (8.6.5) получаем

$$\oint_{\Gamma} \left(u \frac{\partial u}{\partial y} dx - u \frac{\partial u}{\partial x} dy \right) - \iint_D \left[\left(\frac{\partial u}{\partial x} \right)^2 + \left(\frac{\partial u}{\partial y} \right)^2 \right] dx dy = 0. \quad (8.6.6)$$

В силу условий (8.6.3) и (8.6.4) функция u равна нулю на контуре Γ , поэтому в (8.6.6) остается только интеграл по области D :

$$\iint_D \left[\left(\frac{\partial u}{\partial x} \right)^2 + \left(\frac{\partial u}{\partial y} \right)^2 \right] dx dy = 0.$$

Из последнего равенства следует $\frac{\partial u}{\partial x} \equiv \frac{\partial u}{\partial y} \equiv 0$. Поэтому $u(x, y) = \text{const}$, и так как на боковых сторонах эта функция обращается в нуль, то $u(x, y) \equiv 0$ в D . Вспоминая определение u , находим $\bar{u}(x, y) \equiv \tilde{u}(x, y)$, что и требовалось доказать.

Для нахождения приближенного решения задачи (8.6.1), (8.3.7) – (8.3.8) используем метод конечных разностей. Введем такую же расчетную сетку в области D , как и в предыдущих параграфах. Для аппроксимации производных от решения во внутренних узлах (x_i, y_j) , $i = 1, 2, \dots, n-1$, $j = 1, 2, \dots, m-1$ области D используем формулы:

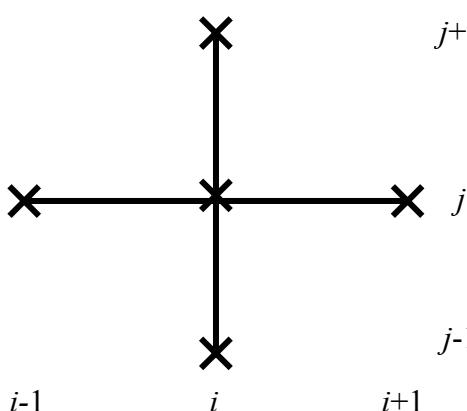
$$\frac{\partial^2 u}{\partial x^2} \Big|_{i,j} = \frac{u(x_{i+1}, y_j) - 2u(x_i, y_j) + u(x_{i-1}, y_j)}{h_1^2} + O(h_1^2), \quad (8.6.7)$$

$$\frac{\partial^2 u}{\partial y^2} \Big|_{i,j} = \frac{u(x_i, y_{j+1}) - 2u(x_i, y_j) + u(x_i, y_{j-1})}{h_2^2} + O(h_2^2). \quad (8.6.8)$$

Подставляя (8.6.7) и (8.6.8) в уравнение (8.6.1), находим

$$\frac{u(x_{i+1}, y_j) - 2u(x_i, y_j) + u(x_{i-1}, y_j)}{h_1^2} + \frac{u(x_i, y_{j+1}) - 2u(x_i, y_j) + u(x_i, y_{j-1})}{h_2^2} + O(h_1^2 + h_2^2) = f(x_i, y_j). \quad (8.6.9)$$

При этом используется шаблон, представленный на рисунке.



Отбрасывая неизвестную погрешность $O(h_1^2 + h_2^2)$, получаем приближенное уравнение

$$\frac{u_{i+1,j} - 2u_{i,j} + u_{i-1,j}}{h_1^2} + \frac{u_{i,j+1} - 2u_{i,j} + u_{i,j-1}}{h_2^2} = f(x_i, y_j), \quad (8.6.10)$$

где $i = 1, 2, \dots, n-1$, $j = 1, 2, \dots, m-1$. К этому уравнению следует присоединить условия, которые вытекают из (8.6.7) и (8.6.8):

$$u_{0,j} = \psi_1(y_j), \quad u_{n,j} = \psi_2(y_j), \quad j = 0, 1, \dots, m, \quad (8.6.11)$$

$$u_{i,0} = \psi_3(x_i), \quad u_{i,m} = \psi_4(x_i), \quad i = 0, 1, \dots, n. \quad (8.6.12)$$

Система алгебраических уравнений (8.6.10) содержит $(n-1)(m-1)$ неизвестных и столько же уравнений. В отличие от подразд. 8.4 и 8.5 здесь не удается получить решение путем последовательного его нахождения на слоях $x = \text{const}$ или $y = \text{const}$. Вместе с тем можно применить другие методы решения линейных систем уравнений, например метод простых итераций. Для простоты рассмотрим его реализацию в частном случае, когда $h_1 = h_2 = h$. Тогда соотношения (8.6.10) принимают вид

$$u_{i,j+1} - 2u_{i,j} + u_{i,j-1} + u_{i+1,j} - 2u_{i-1,j} + u_{i-1,j} = h^2 f(x_i, y_j),$$

то есть

$$u_{i,j} = \frac{1}{4} [u_{i,j+1} + u_{i,j-1} + u_{i+1,j} + u_{i-1,j} - h^2 f(x_i, y_j)]. \quad (8.6.13)$$

Выбирая некоторое нулевое приближение $u_{i,j}^{(0)}$, удовлетворяющее краевым условиям (8.6.11) и (8.6.12), находим следующие приближения согласно методу простых итераций:

$$u_{i,j}^{(k+1)} = \frac{1}{4} [u_{i,j+1}^{(k)} + u_{i,j-1}^{(k)} + u_{i+1,j}^{(k)} + u_{i-1,j}^{(k)} - h^2 f(x_i, y_j)]. \quad (8.6.14)$$

Система уравнений (8.6.13) записана в том же самом виде, как в подразд. 5.5, а именно, с использованием двух индексов в обозначении неизвестных $u_{i,j}$. Можно показать, что справедлива теорема, аналогичная теореме 5.4, поэтому последовательность приближений решения (8.6.14) системы алгебраических уравнений сходится к точному решению этой системы:

$$\|u_{i,j}^{(k)} - u_{i,j}\|_\infty = \max_{i,j} |u_{i,j}^{(k)} - u_{i,j}| \rightarrow 0 \text{ при } k \rightarrow \infty.$$

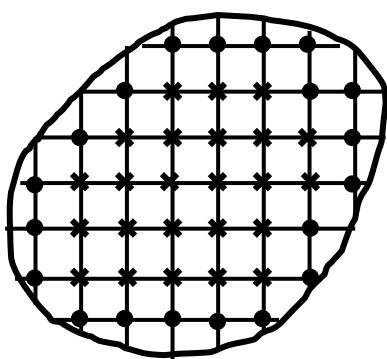
Аналогичным образом может быть найдено решение системы (8.6.10) в случае разных шагов сетки h_1 и h_2 . Для полученного решения $u_{i,j}$ системы уравнений (8.6.10) доказана следующая оценка:

$$\|u_{i,j} - u(x_i, y_j)\|_\infty = O(h_1^2 + h_2^2),$$

из которой следует сходимость $u_{i,j}$ к решению $u(x_i, y_j)$ задачи Дирихле для уравнения (8.6.1) при неограниченном измельчении сетки: $h_1 \rightarrow 0, h_2 \rightarrow 0$.

Если область D не является прямоугольником, то задача Дирихле состоит в нахождении функции $u(x, y)$, являющейся решением уравнения (8.6.1) в области D и принимающей заданные значения $u|_\Gamma = \psi(x, y)$ на ее границе Γ . В этом случае алгоритм решения задачи методом конечных разностей в целом сохраняется, однако возникают значительные трудности, связанные с аппроксимацией производных в приграничных узлах. Так как расстояния от

этих узлов до узлов, расположенных непосредственно на границе, вообще говоря, разные (см. рисунок), то стандартные формулы (8.6.7), (8.6.8) для аппроксимации вторых производных оказываются несправедливыми и возникает необходимость запоминания координат всех граничных узлов, что значительно усложняет составление вычислительной программы. Чтобы избежать этих трудностей, в исходном уравнении, если возможно, используют замену независимых переменных, то есть переход к криволинейной системе координат, в которой расчетной областью является прямоугольник.



8.7. Свойства разностных схем для дифференциальных уравнений: способность аппроксимировать исходную дифференциальную задачу, устойчивость и сходимость

Общая схема решения задач для дифференциальных уравнений методом конечных разностей, которая была использована выше, состоит в следующем. Пусть задан некоторый дифференциальный оператор второго порядка Lu , который является линейным, то есть обладает свойством $L(c_1u_1 + c_2u_2) = c_1Lu_1 + c_2Lu_2$, справедливым для любых дважды непрерывно дифференцируемых функций u_1, u_2 и действительных чисел c_1, c_2 . Для решения уравнения

$$Lu = f \quad (8.7.1)$$

в области его определения D вводится прямоугольная вычислительная сетка и дифференциальный оператор Lu аппроксимируется конечно-разностным оператором $L_{h\tau}u$, так что для точного решения $u(x, t)$, гарантированного теоремой существования, справедливо равенство $Lu = L_{h\tau}u + \alpha_{h,\tau}$, где $\alpha_{h,\tau}$ - погрешность аппроксимации. Если погрешность $\alpha_{h,\tau}$ стремится к нулю при неограниченном измельчении сетки, то говорят, что разностная схема удовлетворяет свойству аппроксимации дифференциального оператора. В разностных схемах, использованных в подразд. 8.4 – 8.6, погрешность имела вид $O(h^2 + \tau)$ или $O(h^2 + \tau^2)$.

В результате исходное дифференциальное уравнение может быть переписано так

$$L_{h\tau}u(x_i, t_j) + \alpha_{h,\tau} = f_{i,j}, \quad (8.7.2)$$

в узлах сетки, где $f_{i,j} = f(x_i, t_j)$. Опуская погрешность, получаем систему алгебраических уравнений

$$L_{h\tau}u_{i,j} = f_{i,j} \quad (8.7.3)$$

относительно приближенного решения $u_{i,j}$.

Для оценки точности приближенного решения важное значение имеет свойство устойчивости решения системы уравнений (8.7.3), которое состоит в следующем. Пусть $\Delta f_{i,j}$ - некоторое возмущение правой части системы и $(u_{i,j} + \Delta u_{i,j})$ - соответствующее ему решение системы

$$L_{h\tau}(u_{i,j} + \Delta u_{i,j}) = f_{i,j} + \Delta f_{i,j}. \quad (8.7.4)$$

Вычитая (8.7.3) из (8.7.4), получаем $L_{h\tau}(u_{i,j} + \Delta u_{i,j}) - L_{h\tau}u_{i,j} = \Delta f_{i,j}$. В силу линейности оператора $L_{h\tau}u$ отсюда находим $L_{h\tau}\Delta u_{i,j} = \Delta f_{i,j}$. Свойство устойчивости записывается в виде

$$\|\Delta u_{i,j}\| \leq c\|\Delta f_{i,j}\| \text{ или } \|\Delta u_{i,j}\| \leq c\|L_{h\tau}\Delta u_{i,j}\|, \quad (8.7.5)$$

где $\|\cdot\|$ - выбранные нормы в конечномерных пространствах, которым принадлежат искомое решение и правая часть системы, а постоянная c не зависит от $f_{i,j}$ и зависит только от коэффициентов оператора $L_{h\tau}u$. Устойчивость решения означает малые его изменения при малых изменениях правой части системы. Покажем, что свойства аппроксимации и устойчивости обеспечивают сходимость приближенного решения к точному: аппроксимация + устойчивость \Rightarrow сходимость.

Действительно, для разности $u - u_{i,j}$ где $u(x_i, t_j)$ - значения решения уравнения (8.7.1) в узлах сетки, а $u_{i,j}$ - решение системы (8.7.3), благодаря свойству устойчивости (8.7.5) получаем

$$\|u - u_{i,j}\| \leq c\|L_{h\tau}(u - u_{i,j})\| = c\|L_{h\tau}u - L_{h\tau}u_{i,j}\|.$$

Теперь, добавляя и вычитая в правой части Lu , преобразуем ее следующим образом:

$$\begin{aligned} \|u - u_{i,j}\| &\leq c\|L_{h\tau}(u - u_{i,j})\| = c\|L_{h\tau}u - L_{h\tau}u_{i,j}\| = c\|L_{h\tau}u - L_{h\tau}u_{i,j} - Lu + Lu\| = \\ &= c\|L_{h\tau}u - Lu + f - f_{i,j}\| = c\|L_{h\tau}u - Lu\| = c\|\alpha_{h,\tau}\| \rightarrow 0 \text{ при } \alpha_{h,\tau} \rightarrow 0, \end{aligned}$$

так как $f - f_{i,j} = 0$ в узлах сетки.

8.8. Некоторые обобщения

Линейные уравнения параболического, гиперболического и эллиптического типов с переменными коэффициентами согласно (8.3.1), (8.3.4), (8.3.6) имеют вид

$$\frac{\partial u}{\partial t} - \alpha(x, t) \frac{\partial^2 u}{\partial x^2} + d(x, t) \frac{\partial u}{\partial x} + e(x, t) \frac{\partial u}{\partial t} + g(x, t)u = f(x, t), \quad (8.8.1)$$

$$\frac{\partial^2 u}{\partial t^2} - \alpha(x, t) \frac{\partial^2 u}{\partial x^2} + d(x, t) \frac{\partial u}{\partial x} + e(x, t) \frac{\partial u}{\partial t} + g(x, t)u = f(x, t), \quad (8.8.2)$$

$$\frac{\partial^2 u}{\partial x^2} + \frac{\partial^2 u}{\partial y^2} + d(x, t) \frac{\partial u}{\partial x} + e(x, t) \frac{\partial u}{\partial y} + g(x, t)u = f(x, t), \quad (8.8.3)$$

где $\alpha(x, t) > 0$. Решение краевых задач для уравнений (8.8.1) – (8.8.3) может осуществляться с помощью тех же разностных схем, которые описаны в подразд. 8.4 – 8.6, хотя в некоторых случаях могут возникать определенные трудности, например, в случае уравнения параболического типа (8.8.1) с преобладающим влиянием конвективного слагаемого $d(x, t) \frac{\partial u}{\partial x}$ по

сравнению с диффузионным слагаемым $\alpha(x, t) \frac{\partial^2 u}{\partial x^2}$ (см. (8.4.20)). Отметим также важное отличие уравнения эллиптического типа (8.8.3) от первых двух, состоящее в необходимости выполнения условия $g(x, y) \leq 0$ в области D , которое обеспечивает существование и единственность точного решения краевой задачи. При нарушении этого условия корректность исходной задачи может нарушаться.

В качестве краевых условий могут быть заданы не значения решения на границе рассматриваемой области, а значения его производной по нормали к границе. Например, для уравнения (8.8.1) может быть рассмотрена задача в прямоугольнике $D = \left\{ (x, t) / 0 < x < l, 0 < t < T \right\}$ с начальными условиями (8.3.2) и краевыми условиями

$$\left. \frac{\partial u}{\partial x} \right|_{x=0} = \psi_1(t), \quad \left. \frac{\partial u}{\partial x} \right|_{x=l} = \psi_2(t), \quad 0 \leq t \leq T,$$

которые заменяют условия (8.3.3). Решение такой задачи может быть получено с помощью тех же разностных схем, которые построены в подразд. 8.4. Действительно, значение приближенного решения во внутренних узлах $i = 1, 2, \dots, n-1$ слоя $t = t_{j+1}$ может быть найдено, например, по формуле (8.4.9). Значения решения в левой граничной точке $u_{0,j+1}$ могут быть получены из соотношения для трехточечной аппроксимации первой производной:

$$\left. \frac{\partial u}{\partial x} \right|_{x=0} \approx \frac{-3u_{0,j+1} + 4u_{1,j+1} - u_{2,j+1}}{2h}, \quad \text{погрешность которой имеет порядок } O(h^2).$$

Аналогично,

значения решения в правой граничной точке $u_{n,j+1}$ могут быть найдены из соотношения

$$\left. \frac{\partial u}{\partial x} \right|_{x=l} \approx \frac{3u_{n,j+1} - 4u_{n-1,j+1} + u_{n-2,j+1}}{2h}.$$

Поэтому значения приближенного решения во всех узлах, расположенных на слое $t = t_{j+1}$, оказываются известными, и можно переходить к нахождению решения на слое $t = t_{j+2}$.

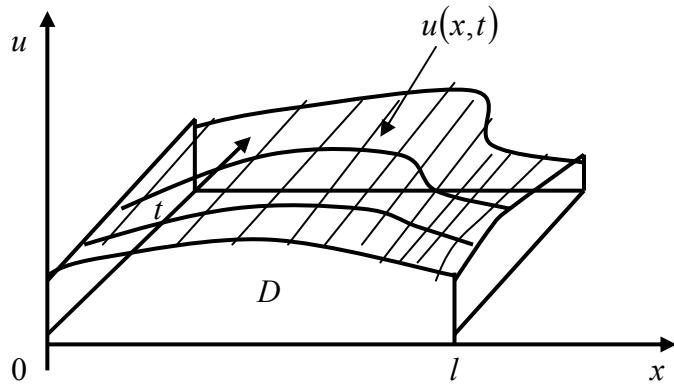
Примерами нелинейных уравнений параболического, гиперболического и эллиптического типов являются:

$$\frac{\partial u}{\partial t} - \alpha(x, t, u) \frac{\partial^2 u}{\partial x^2} = f\left(x, t, u, \frac{\partial u}{\partial x}, \frac{\partial u}{\partial t}\right), \quad (8.8.4)$$

$$\frac{\partial^2 u}{\partial t^2} - \alpha(x, t, u) \frac{\partial^2 u}{\partial x^2} = f(x, t, u, \frac{\partial u}{\partial x}, \frac{\partial u}{\partial t}), \quad (8.8.5)$$

$$\frac{\partial^2 u}{\partial x^2} + \alpha(x, y, u) \frac{\partial^2 u}{\partial y^2} = f(x, y, u, \frac{\partial u}{\partial x}, \frac{\partial u}{\partial y}), \quad (8.8.6)$$

где $\alpha > 0$. Решение задач для уравнений (8.8.4) – (8.8.6) может производиться по тем же алгоритмам, которые представлены в подразд. 8.4 – 8.6. Однако важным свойством нелинейных уравнений гиперболического типа (8.8.5) является возможность возникновения разрывов решений даже при гладких начальных данных в результате нарастания величины $|\frac{\partial u}{\partial x}|$ с



ростом t , то есть в результате увеличения крутизны волн, описываемых эти решениями и распространяющимися вдоль оси x (см. рисунок). В этих случаях вопрос о сходимости приближенных решений алгебраических уравнений (8.8.5) остается открытым, поэтому построение разностных схем и проверка достоверности полученного результата проводится обычно на уровне вычислительного эксперимента.

Примерами уравнений параболического, гиперболического и эллиптического типов с тремя независимыми переменными являются:

$$\frac{\partial u}{\partial t} - \alpha(x, y, t) \left(\frac{\partial^2 u}{\partial x^2} + \frac{\partial^2 u}{\partial y^2} \right) = f(x, y, t), \quad (8.8.7)$$

$$\frac{\partial^2 u}{\partial t^2} - \alpha(x, y, t) \left(\frac{\partial^2 u}{\partial x^2} + \frac{\partial^2 u}{\partial y^2} \right) = f(x, y, t), \quad (8.8.8)$$

$$\frac{\partial^2 u}{\partial x^2} + \frac{\partial^2 u}{\partial y^2} + \frac{\partial^2 u}{\partial z^2} = f(x, y, z), \quad (8.8.9)$$

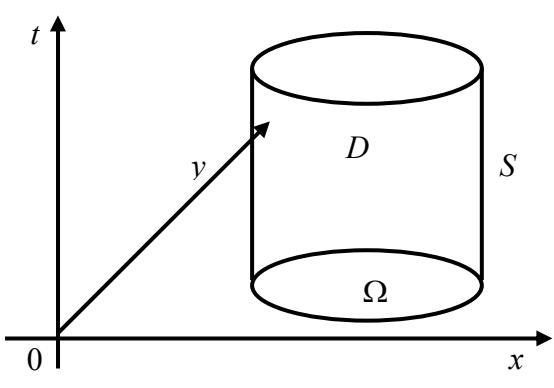
где $\alpha(x, y, t) > 0$. Уравнение (8.8.7), например, описывает распространение тепла в твердом теле в случае, когда определяющее значение имеет теплопередача в двух пространственных направлениях, а в третьем направлении потока тепла нет.

Уравнение (8.8.8) описывает колебания упругой натянутой мембраны, аналогичные колебаниям струны, описанных в подразд. 8.5. Рассмотрим формулировку задачи о колебаниях мембранны подробнее.

Пусть Ω – некоторая область, расположенная в плоскости (x, y) , с границей Γ . Обозначим через D цилиндрическую область $D = \{(x, y, t) / (x, y) \in \Omega, 0 < t < T\}$ (см. рисунок).

Требуется найти решение уравнения (8.8.8), удовлетворяющего следующим начальным и краевому условиям:

$$u(x, y, 0) = \varphi_0(x, t), \quad \frac{\partial u}{\partial t} \Big|_{t=0} = \varphi_1(x, y), \quad (x, y) \in \Omega, \quad (8.8.10)$$



$$u|_S = \psi(x, y, t), \quad (8.8.11)$$

где S - боковая граница области D . Правая часть $f(x, y, t)$ в уравнении (8.8.8) имеет физический смысл внешней силовой нагрузки, распределенной по поверхности мембраны, занимающей область Ω . Если $\alpha(x, y, t) = \text{const}$ и $\psi \equiv 0$, то мембрана является однородной и края ее закреплены неподвижно.

Для нахождения приближенного решения задачи (8.8.8), (8.8.10), (8.8.11) воспользуемся методом конечных разностей. Введем в области D трехмерную прямоугольную расчетную сетку (x_i, y_j, t_k) , $x_i = h_1 i$, $y_j = h_2 j$, $t_k = \tau k$. Для аппроксимации производных от решения по x и y в узлах (x_i, y_j, t_k) используем формулы (8.6.7) и (8.6.8), а для аппроксимации производной $\frac{\partial^2 u}{\partial t^2}$ - формулу вида (8.5.3). Тогда, отбрасывая погрешности аппроксимации, из (8.8.8) получаем соотношение

$$\frac{u_{i,j,k+1} - 2u_{i,j,k} + u_{i,j,k-1}}{\tau^2} - \alpha(x_i, y_j, t_k) \left(\frac{u_{i+1,j,k} - 2u_{i,j,k} + u_{i-1,j,k}}{h_1^2} + \frac{u_{i,j+1,k} - 2u_{i,j,k} + u_{i,j-1,k}}{h_2^2} \right) = f(x_i, y_j, t_k),$$

которое связывает значения решения в пяти соседних узлах, расположенных на слое $t = t_k$, и в двух узлах, расположенных на слоях $t = t_{k+1}$ и $t = t_{k-1}$. Полученное соотношение дает явное представление для значений $u_{i,j,k+1}$ на слое $t = t_{k+1}$ и позволяет последовательно вычислить приближенное решение во всех узлах расчетной сетки в области D .

Простейшим примером уравнения с частными производными четвертого порядка с двумя независимыми переменными является уравнение

$$\frac{\partial^2 u}{\partial t^2} - a_0^2 \frac{\partial^4 u}{\partial x^4} = f(x, t), \quad (8.8.12)$$

которое описывает упругие изгибные колебания балки или стержня. Корректная постановка задачи включает в себя, кроме уравнения (8.8.12), два начальных и четыре краевых условия. Для этой задачи можно записать явную разностную схему и последовательно по слоям $t = t_j$ определить приближенное решение во всей расчетной области.

Решение более сложных уравнений порядка выше второго и систем уравнений с частными производными обычно связано со значительными трудностями. Построение разностных схем для таких уравнений и систем, как правило, требует индивидуального подхода. Поэтому в разных областях физики и техники разработаны специальные численные методы, которые ориентированы на изучение адекватных математических моделей сложных физических явлений.

8.9. Лабораторная работа № 14. Решение задачи Дирихле для уравнения Лапласа методом сеток

Найти приближенное решение уравнения Лапласа

$$\Delta u \equiv \frac{\partial^2 u}{\partial x^2} + \frac{\partial^2 u}{\partial y^2} = 0 \quad (8.9.1)$$

в квадрате $D = \{(x, y) / 0 \leq x \leq l_1, 0 \leq y \leq l_2\}$, принимающее на границе области D заданные краевые условия (8.3.7) и (8.3.8):

$$\begin{aligned} u(0, y) &= \psi_1(y), & u(l_1, y) &= \psi_2(y), & 0 \leq y \leq l_2, \\ u(x, 0) &= \psi_3(x), & u(x, l_2) &= \psi_4(x), & 0 \leq x \leq l_1 \end{aligned} \quad (8.9.2)$$

(см. рисунок на с. 198).

Построим область решения, покроем ее сеткой с шагом $h_1 = h_2 = h$, $x_i = i \cdot h$, $y_j = j \cdot h$, $i = 0, 1, \dots, n$, $j = 0, 1, \dots, m$ и вычислим значения искомой функции $u(x, y)$ в граничных точках области по формулам (8.9.2). Введем обозначения $u_{i,j} = u(x_i, y_j)$ и аппроксимируем

частные производные $\frac{\partial^2 u}{\partial x^2}$ и $\frac{\partial^2 u}{\partial y^2}$ в каждом внутреннем узле сетки центральными разностными производными второго порядка по формулам (8.6.7) и (8.6.8):

$$\begin{aligned}\left. \frac{\partial^2 u}{\partial x^2} \right|_{i,j} &= \frac{u_{i+1,j} - 2u_{i,j} + u_{i-1,j}}{h^2} + O(h^2), \\ \left. \frac{\partial^2 u}{\partial y^2} \right|_{i,j} &= \frac{u_{i,j+1} - 2u_{i,j} + u_{i,j-1}}{h^2} + O(h^2).\end{aligned}\quad (8.9.3)$$

Уравнение Лапласа (8.9.1) заменим конечно-разностным уравнением

$$\frac{u_{i+1,j} - 2u_{i,j} + u_{i-1,j}}{h^2} + \frac{u_{i,j+1} - 2u_{i,j} + u_{i,j-1}}{h^2} = 0. \quad (8.9.4)$$

Погрешность замены дифференциального уравнения разностным составляет величину $O(h^2)$. Уравнение (8.9.4) вместе со значениями $u_{i,j}$ в граничных узлах образуют систему линейных алгебраических уравнений относительно приближенных значений $u(x, y)$ в узлах сетки:

$$u_{i,j} = \frac{1}{4}(u_{i,j+1} + u_{i,j-1} + u_{i+1,j} + u_{i-1,j}). \quad (8.9.5)$$

При получении сеточных уравнений (8.9.5) использовалась схема узлов, изображенная на с. 208.

Численное решение задачи Дирихле для уравнения Лапласа в области D состоит в нахождении приближенных значений $u_{i,j}$ искомой функции $u(x, y)$ во внутренних узлах сетки. Для определения величин $u_{i,j}$ необходимо решить систему линейных алгебраических уравнений (8.9.5).

В данной лабораторной работе система (8.9.5) решается методом простых итераций (см. подразд. 5.5) по формулам

$$u_{i,j}^{(k+1)} = \frac{1}{4}(u_{i,j+1}^{(k)} + u_{i,j-1}^{(k)} + u_{i+1,j}^{(k)} + u_{i-1,j}^{(k)}), \quad (8.9.6)$$

где верхним индексом k обозначен номер итерации. В качестве условия окончания итерационного процесса можно принять

$$\max_{i,j} |u_{i,j}^{(k)} - u_{i,j}^{(k+1)}| < \varepsilon, \quad i = 1, 2, \dots, n-1, \quad j = 1, 2, \dots, m-1. \quad (8.9.7)$$

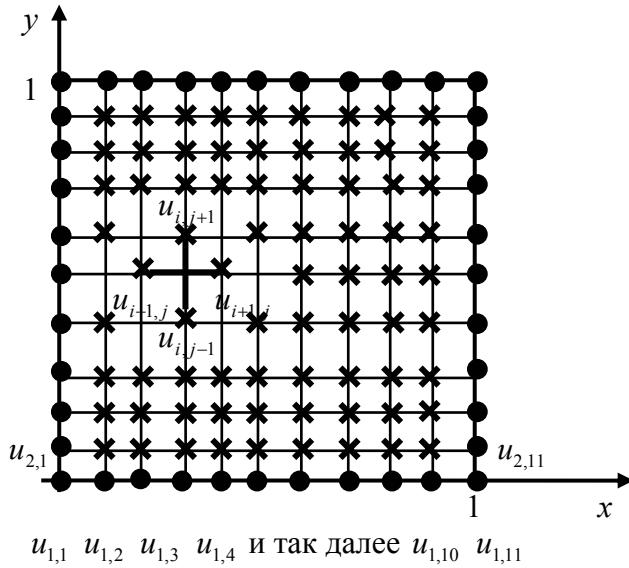
Как следует из теории, изложенной в подразд. 8.6, описанная разностная схема обладает свойством устойчивости и сходимости. Это означает, что, выбрав достаточно малый шаг h , можно сколь угодно точно решить исходную задачу.

Итак, решим по описанной процедуре модельный пример. Пусть надо найти решение $u(x, y)$ задачи Дирихле в квадрате со стороной, равной единице, для уравнения Лапласа (8.9.1) с краевыми условиями вида

$$u(0, y) = \psi_1(y) = y^2, \quad u(1, y) = \psi_2(y) = \cos y + y(3 - \cos 1),$$

$$u(x, 0) = \psi_3(x) = x^3, \quad u(x, 1) = \psi_4(x) = 2x + 1, \quad 0 \leq x \leq 1, \quad 0 \leq y \leq 1.$$

Зададим одинаковый шаг по x и по y , равный $h = 0.1$, и вычислим все краевые условия. На приведенном рисунке известные значения функции $u(x, y)$ в граничных точках помечены жирными точками, разыскиваемые приближенные решения внутри квадрата D - крестиками, внутри D выделен используемый шаблон.



чены жирными точками, разыскиваемые приближенные решения внутри квадрата D - крестиками, внутри D выделен используемый шаблон.

ORIGIN:=1

$$\begin{aligned}\psi 1(y) &:= y^2 & \psi 2(y) &:= \cos(y) + y * (3 - \cos(1)) \\ \psi 3(x) &:= x^3 & \psi 4(x) &:= 2 * x + 1 \\ n &:= 11 & m &:= 11 & h &:= 0.1 \\ i &:= 1 \dots n & x_i &:= (i-1) * h & u_{i,1} &:= \psi 3(x_i) & u_{i,n} &:= \psi 4(x_i) \\ j &:= 1 \dots m & y_j &:= (j-1) * h & u_{1,j} &:= \psi 1(y_j) & u_{m,j} &:= \psi 2(y_j)\end{aligned}$$

Поскольку система (8.9.5) будет решаться методом простых итераций, зададим начальное приближение, то есть вектор $\bar{u}^{(0)}$ в каждом горизонтальном слое. Сначала рассмотрим горизонталь с граничными точками $(0, 0.1)$ и $(1, 0.1)$. Будем считать, что функция $u(x, y)$ по горизонталям области D распределена равномерно. $u_{2,1} = \psi 1(0, 1) = 0.01$, $u_{2,11} = \psi 2(0, 1) = 1.241$, и так как отрезок разбит на десять частей (см. рисунок), то шаг изме-

$$u_{2,1} = 0.01 \quad \bar{u}_{2,3}^{(0)} \quad \bar{u}_{2,5}^{(0)} \quad \bar{u}_{2,7}^{(0)} \quad \bar{u}_{2,9}^{(0)} \quad u_{2,11} = 1.241$$

нения функции $l = (1.241 - 0.01) / 10 = 0.12$.

Тогда получим $\bar{u}_{2,2}^{(0)} = u_{2,1} + l = 0.13$, $\bar{u}_{2,3}^{(0)} = u_{2,1} + 2l = 0.25$ и так далее. Аналогичным образом рассчитаем начальные значения функции во внутренних точках других горизонталей.

Выполним теперь эти же действия средствами пакета Mathcad:

$$i := 2 \dots n-1 \quad j := 2 \dots m-1$$

$$l_j := h * (u_{n,j} - u_{1,j}) \quad u_{i,j} := u_{1,j} + (i-1)*l_j$$

	1	2	3	4	5	6	7	8	9	10	11
1	0	0.01	0.04	0.09	0.16	0.25	0.36	0.49	0.64	0.81	1
2	0.001	0.133097	0.183201	0.250325	0.334494	0.435743	0.554115	0.689663	0.842446	1.012534	1.2
3	0.008	0.256195	0.326401	0.410649	0.508988	0.621486	0.748231	0.889326	1.044893	1.215068	1.4
4	0.027	0.379292	0.469602	0.570974	0.683482	0.807229	0.942346	1.088989	1.247339	1.417601	1.6
5	0.064	0.50239	0.612802	0.731298	0.857976	0.992973	1.136462	1.288652	1.449786	1.620135	1.8
u =	6	0.125	0.625487	0.756003	0.891623	1.03247	1.178716	1.330577	1.488315	1.652232	1.822669
7	0.216	0.748584	0.899204	1.051947	1.206964	1.364459	1.524693	1.687978	1.854679	2.025203	2.2
8	0.343	0.871682	1.042404	1.212272	1.381458	1.550202	1.718808	1.887641	2.057125	2.227737	2.4
9	0.512	0.994779	1.185605	1.372597	1.555952	1.735945	1.912923	2.087304	2.259572	2.43027	2.6
10	0.729	1.117877	1.328806	1.532921	1.730446	1.921688	2.107039	2.286968	2.462018	2.632804	2.8
11	1	1.240974	1.472006	1.693246	1.90494	2.107431	2.301154	2.486631	2.664465	2.835338	3

Теперь можно запустить вычисления по методу простых итераций. На практике вместо условий (8.9.7) окончания итерационного процесса применяют более надежный критерий

$$\max_{i,j} |u_{i,j}^{(k+1)} - u_{i,j}^{(k)}| \leq \varepsilon(1-\nu), \quad (8.9.8)$$

где $\nu = \frac{\max_{i,j} |u_{i,j}^{(k+1)} - u_{i,j}^{(k)}|}{\max_{i,j} |u_{i,j}^{(k)} - u_{i,j}^{(k-1)}|}$. Метод простых итераций выгоднее всего оформить в виде небольшой подпрограммы. Попытка организовать вычисления с помощью дискретных аргументов приведет к неоправданной сложности математических выражений. Записать процесс последовательных приближений в векторно-матричной форме, как в лабораторной работе № 9 (см. с. 137), тоже не удастся, так как необходимо обеспечить выход из итерационного процесса по достижении заданной точности.

Программа метода простых итераций очень проста и понятна и может быть, например, такой:

```

Iteratio(u, eps) := | n←rows(u)
                     m←cols(u)
                     r2←1010
                     te←0
                     k←0
                     while r2>te
                           r1←r2
                           r2←0
                           for i∈2..n-1
                               for j∈2..m-1
                                   x← $\frac{1}{4} \cdot (u_{i,j+1} + u_{i,j-1} + u_{i+1,j} + u_{i-1,j})$ 
                                   r←|x - u_{i,j}|
                                   r2←r if r2≤r
                                   u_{i,j}←x
                           te←eps· $\left(1 - \frac{r2}{r1}\right)$ 
                           k←k+1
                     return k if k>1000
u

```

$\text{eps} := 10^{-6}$ $u := \text{Iteratio}(u, \text{eps})$

	1	2	3	4	5	6	7	8	9	10	11
$u =$	1	0	0.01	0.04	0.09	0.16	0.25	0.36	0.49	0.64	0.81
	2	0.001	0.073388	0.150422	0.23665	0.334429	0.445138	0.569583	0.708099	0.860472	1.025543
	3	0.008	0.132131	0.251651	0.371749	0.495927	0.626541	0.765093	0.912342	1.068246	1.2317
	4	0.027	0.195485	0.352303	0.502766	0.65099	0.800007	0.951906	1.107928	1.26847	1.433009
	5	0.064	0.270506	0.459309	0.636022	0.80526	0.970589	1.134596	1.298996	1.464698	1.631868
	6	0.125	0.36323	0.578403	0.776754	0.963441	1.142491	1.316895	1.48876	1.659458	1.829763
	7	0.216	0.479011	0.714321	0.929148	1.129258	1.319039	1.501732	1.679691	1.854611	2.027727
	8	0.343	0.622494	0.87072	1.096258	1.305406	1.502675	1.691303	1.873661	2.051566	2.226534
	9	0.512	0.797246	1.049807	1.279757	1.493433	1.694952	1.887143	2.072084	2.25146	2.426844
	10	0.729	1.004682	1.251507	1.479531	1.693615	1.896558	2.090232	2.276073	2.455345	2.629382
	11	1	1.240974	1.472006	1.693246	1.90494	2.107431	2.301154	2.486631	2.664465	2.835338

Число итераций для выполнения условия сходимости (8.9.8) в этом примере равно 111.

Наконец, описанный выше алгоритм можно оформить в виде единственной подпрограммы Dirichlet:

```
Dirichlet(a, b, c, d, n, eps) := 
  11 ← b - a
  12 ← d - c
  h ←  $\frac{11}{(n - 1)}$ 
  m ←  $\frac{12}{h} + 1$ 
  for i ∈ 1..n
    xi ← (i - 1) · h
    ui,1 ← ψ3(xi)
    ui,n ← ψ4(xi)
  for j ∈ 1..m
    yj ← (j - 1) · h
    u1,j ← ψ1(yj)
    um,j ← ψ2(yj)
  for i ∈ 2..n - 1
    for j ∈ 2..m - 1
      x ← h · (un,j - u1,j)
      ui,j ← u1,j + (i - 1) · x
  u ← Iteratio(u, eps)
  u
```

Входные параметры этой подпрограммы: $[a, b]$ - отрезок по оси x , $[c, d]$ - отрезок по оси y , n - число шагов по x . Так как использована формула (8.9.5), где $h_1 = h_2 = h$, то число

шагов сетки по оси y вычисляется в самой программе. Желающие могут легко усовершенствовать алгоритм для случая $h_1 \neq h_2$.

Итак, решим еще раз уравнение Лапласа для этого примера.

$u := \text{Dirichlet}(0, 1, 0, 1, 11, \text{eps})$

Если распечатать еще раз матрицу u , получим точно такую же таблицу, как на предыдущей странице. Как видно, результаты вычислений полностью совпадают.

Задание № 1. С помощью подпрограммы Dirichlet найти приближенное решение уравнения Лапласа в заданной области D с указанными граничными условиями.

$$1. \quad D = \left\{ \begin{array}{l} (x, y) \\ \diagup 0 \leq x \leq 1, 0 \leq y \leq 1 \end{array} \right\},$$

$$u(0, y) = \psi_1(y) = 20y, \quad u(1, y) = \psi_2(y) = 20 \cos \frac{\pi y}{2},$$

$$u(x, 0) = \psi_3(x) = 20x^2, \quad u(x, 1) = \psi_4(x) = 20 \cos \frac{\pi x}{2},$$

$$h_1 = h_2 = h = 0.1.$$

$$2. \quad D = \left\{ \begin{array}{l} (x, y) \\ \diagup -1 \leq x \leq 1, 0 \leq y \leq 1 \end{array} \right\},$$

$$u(-1, y) = \psi_1(y) = -10, \quad u(1, y) = \psi_2(y) = 5y(1 - y^2),$$

$$u(x, 0) = \psi_3(x) = 5x(1 - x), \quad u(x, 1) = \psi_4(x) = 5x(1 - x),$$

$$h_1 = h_2 = h = 0.1.$$

$$3. \quad D = \left\{ \begin{array}{l} (x, y) \\ \diagup 0 \leq x \leq 1, 0 \leq y \leq 1 \end{array} \right\},$$

$$u(0, y) = \psi_1(y) = 30(1 - y), \quad u(1, y) = \psi_2(y) = 20y,$$

$$u(x, 0) = \psi_3(x) = 30(1 - x), \quad u(x, 1) = \psi_4(x) = 20\sqrt{x},$$

$$h_1 = h_2 = h = 0.1.$$

$$4. \quad D = \left\{ \begin{array}{l} (x, y) \\ \diagup 0 \leq x \leq \frac{\pi}{2}, 0 \leq y \leq \frac{\pi}{2} \end{array} \right\},$$

$$u(0, y) = \psi_1(y) = 50 \cos y, \quad u\left(\frac{\pi}{2}, y\right) = \psi_2(y) = 30y^2,$$

$$u(x, 0) = \psi_3(x) = 50 \cos x, \quad u\left(x, \frac{\pi}{2}\right) = \psi_4(x) = 30x^2,$$

$$h_1 = h_2 = h = \frac{\pi}{20}.$$

$$5. \quad D = \left\{ \begin{array}{l} (x, y) \\ \diagup 0 \leq x \leq 2, 0 \leq y \leq 2 \end{array} \right\},$$

$$u(0, y) = \psi_1(y) = 20 \cos \frac{\pi y}{2}, \quad u(2, y) = \psi_2(y) = 3y(1 - y^2),$$

$$u(x, 0) = \psi_3(x) = 1 + 1.5x - x^2, \quad u(x, 2) = \psi_4(x) = x - 20,$$

$$h_1 = h_2 = h = 0.2.$$

$$6. \quad D = \left\{ \begin{array}{l} (x, y) \\ \diagup -1 \leq x \leq 0, -1 \leq y \leq 0 \end{array} \right\},$$

$$u(-1, y) = \psi_1(y) = 20 \sin \pi y, \quad u(0, y) = \psi_2(y) = 30y(1 + y),$$

$$u(x, -1) = \psi_3(x) = 20x(1 + x), \quad u(x, 0) = \psi_4(x) = 30x(1 + x),$$

$$h_1 = h_2 = h = 0.1.$$

7. $D = \left\{ \begin{array}{l} (x, y) \\ \diagup 0 \leq x \leq 1, 0 \leq y \leq 1 \end{array} \right\},$

$$u(0, y) = \psi_1(y) = e^y - ey^2, \quad u(1, y) = \psi_2(y) = y,$$

$$u(x, 0) = \psi_3(x) = -x^3 + 1, \quad u(x, 1) = \psi_4(x) = x^2,$$

$$h_1 = h_2 = h = 0.1.$$

8. $D = \left\{ \begin{array}{l} (x, y) \\ \diagup 0 \leq x \leq 1, -1 \leq y \leq 1 \end{array} \right\},$

$$u(0, y) = \psi_1(y) = -y^2 + 1, \quad u(1, y) = \psi_2(y) = y + 1,$$

$$u(x, -1) = \psi_3(x) = \sin x - x^3 \sin 1, \quad u(x, 1) = \psi_4(x) = x(x + 1),$$

$$h_1 = h_2 = h = 0.2.$$

9. $D = \left\{ \begin{array}{l} (x, y) \\ \diagup -1 \leq x \leq 1, 0 \leq y \leq 1 \end{array} \right\},$

$$u(-1, y) = \psi_1(y) = 0, \quad u(1, y) = \psi_2(y) = 2y,$$

$$u(x, 0) = \psi_3(x) = \sin x - x^3 \sin 1, \quad u(x, 1) = \psi_4(x) = x + 1,$$

$$h_1 = h_2 = h = 0.1.$$

10. $D = \left\{ \begin{array}{l} (x, y) \\ \diagup 0 \leq x \leq 1, 0 \leq y \leq 1 \end{array} \right\},$

$$u(0, y) = \psi_1(y) = 2ey - (1 + 2e)y^2 - 1, \quad u(1, y) = \psi_2(y) = -y,$$

$$u(x, 0) = \psi_3(x) = x^3 - 1, \quad u(x, 1) = \psi_4(x) = x - 2,$$

$$h_1 = h_2 = h = 0.1.$$

11. $D = \left\{ \begin{array}{l} (x, y) \\ \diagup 0 \leq x \leq 1, 0 \leq y \leq 1 \end{array} \right\},$

$$u(0, y) = \psi_1(y) = -2y - 4y^2, \quad u(1, y) = \psi_2(y) = 4 - 12y - 4y^2,$$

$$u(x, 0) = \psi_3(x) = x + 3x^2, \quad u(x, 1) = \psi_4(x) = -6 - 9x + 3x^2,$$

$$h_1 = h_2 = h = 0.1.$$

12. $D = \left\{ \begin{array}{l} (x, y) \\ \diagup 1 \leq x \leq 2, 1 \leq y \leq 2 \end{array} \right\},$

$$u(1, y) = \psi_1(y) = e^{y-1}, \quad u(2, y) = \psi_2(y) = e^y,$$

$$u(x, 1) = \psi_3(x) = e^{x-1}, \quad u(x, 2) = \psi_4(x) = e^x,$$

$$h_1 = h_2 = h = 0.1.$$

13. $D = \left\{ \begin{array}{l} (x, y) \\ \diagup -1 \leq x \leq 0, -1 \leq y \leq 0 \end{array} \right\},$

$$u(-1, y) = \psi_1(y) = e^{y^2}, \quad u(0, y) = \psi_2(y) = e^{1-y^2},$$

$$u(x, -1) = \psi_3(x) = e^{x^2}, \quad u(x, 0) = \psi_4(x) = e^{1-x^2},$$

$$h_1 = h_2 = h = 0.1.$$

14. $D = \left\{ \begin{array}{l} (x, y) \\ \diagup -\frac{\pi}{2} \leq x \leq \frac{\pi}{2}, -\frac{\pi}{2} \leq y \leq \frac{\pi}{2} \end{array} \right\},$

$$u\left(-\frac{\pi}{2}, y\right) = \psi_1(y) = 1, \quad u\left(\frac{\pi}{2}, y\right) = \psi_2(y) = 1 + \cos y,$$

$$u\left(x, -\frac{\pi}{2}\right) = \psi_3(x) = \frac{2|x|}{\pi}, \quad u\left(x, \frac{\pi}{2}\right) = \psi_4(x) = 1 + \cos x,$$

$$h_1 = h_2 = h = \frac{\pi}{10}.$$

15. $D = \left\{ \begin{array}{l} (x, y) \\ \diagup 0 \leq x \leq 1, 0 \leq y \leq 1 \end{array} \right\},$
 $u(0, y) = \psi_1(y) = y^2 + 4y, \quad u(1, y) = \psi_2(y) = y^2 + 4y + 4,$
 $u(x, 0) = \psi_3(x) = x^2 + 3x, \quad u(x, 1) = \psi_4(x) = x^2 + 3x + 5,$
 $h_1 = h_2 = h = 0.1.$

16. $D = \left\{ \begin{array}{l} (x, y) \\ \diagup 0 \leq x \leq \pi, 0 \leq y \leq \frac{\pi}{2} \end{array} \right\},$
 $u(0, y) = \psi_1(y) = -\sin y, \quad u(\pi, y) = \psi_2(y) = 1 - \cos y,$
 $u(x, 0) = \psi_3(x) = \sin x, \quad u\left(x, \frac{\pi}{2}\right) = \psi_4(x) = -\cos x,$
 $h_1 = h_2 = h = \frac{\pi}{10}.$

17. $D = \left\{ \begin{array}{l} (x, y) \\ \diagup 0 \leq x \leq 1, 0 \leq y \leq 1 \end{array} \right\},$
 $u(0, y) = \psi_1(y) = 40(1-y), \quad u(1, y) = \psi_2(y) = 30y,$
 $u(x, 0) = \psi_3(x) = 40(1-x), \quad u(x, 1) = \psi_4(x) = 30\sqrt{x},$
 $h_1 = h_2 = h = 0.1.$

18. $D = \left\{ \begin{array}{l} (x, y) \\ \diagup -1 \leq x \leq 1, -1 \leq y \leq 1 \end{array} \right\},$
 $u(-1, y) = \psi_1(y) = 20 \sin \pi y, \quad u(1, y) = \psi_2(y) = 50(1 - y^2),$
 $u(x, -1) = \psi_3(x) = 20 \sin \pi x, \quad u(x, 1) = \psi_4(x) = 50\sqrt{1 - |x|},$
 $h_1 = h_2 = h = 0.2.$

19. $D = \left\{ \begin{array}{l} (x, y) \\ \diagup -2 \leq x \leq -1, 0 \leq y \leq 1 \end{array} \right\},$
 $u(-2, y) = \psi_1(y) = 30y^2, \quad u(-1, y) = \psi_2(y) = 0,$
 $u(x, 0) = \psi_3(x) = 40(x+1)^2(2+x), \quad u(x, 1) = \psi_4(x) = 30|1+x|,$
 $h_1 = h_2 = h = 0.1.$

20. $D = \left\{ \begin{array}{l} (x, y) \\ \diagup 0 \leq x \leq 1, 0 \leq y \leq 1 \end{array} \right\},$
 $u(0, y) = \psi_1(y) = 15\sqrt{y}, \quad u(1, y) = \psi_2(y) = 30y(1-y),$
 $u(x, 0) = \psi_3(x) = 0, \quad u(x, 1) = \psi_4(x) = 15(1-x),$
 $h_1 = h_2 = h = 0.1.$

21. $D = \left\{ \begin{array}{l} (x, y) \\ \diagup 1 \leq x \leq 3, 1 \leq y \leq 3 \end{array} \right\},$
 $u(1, y) = \psi_1(y) = 30 \cos \frac{\pi y}{2}, \quad u(3, y) = \psi_2(y) = 25(y-3)(1-y^2),$
 $u(x, 1) = \psi_3(x) = 30(1-x^2), \quad u(x, 3) = \psi_4(x) = 20(x-3)(1-x),$
 $h_1 = h_2 = h = 0.2.$

22. $D = \left\{ \begin{array}{l} (x, y) \\ \diagup 0 \leq x \leq 1, 0 \leq y \leq 1 \end{array} \right\},$
 $u(0, y) = \psi_1(y) = 25y, \quad u(1, y) = \psi_2(y) = 30\sqrt{y}(1-y),$
 $u(x, 0) = \psi_3(x) = 0, \quad u(x, 1) = \psi_4(x) = 25(1-x^2),$
 $h_1 = h_2 = h = 0.1.$
23. $D = \left\{ \begin{array}{l} (x, y) \\ \diagup 0 \leq x \leq 1, 0 \leq y \leq 1 \end{array} \right\},$
 $u(0, y) = \psi_1(y) = -10y^2 - 8y + 6, \quad u(1, y) = \psi_2(y) = -10y^2 - 30y + 22,$
 $u(x, 0) = \psi_3(x) = 9x^2 + 7x + 6, \quad u(x, 1) = \psi_4(x) = 9x^2 - 15x - 12,$
 $h_1 = h_2 = h = 0.1.$
24. $D = \left\{ \begin{array}{l} (x, y) \\ \diagup 0 \leq x \leq 1, 0 \leq y \leq 1 \end{array} \right\},$
 $u(0, y) = \psi_1(y) = -6y^2 - 4y + 2, \quad u(1, y) = \psi_2(y) = -6y^2 - 18y + 10,$
 $u(x, 0) = \psi_3(x) = 5x^2 + 3x + 2, \quad u(x, 1) = \psi_4(x) = 5x^2 - 11x - 8,$
 $h_1 = h_2 = h = 0.1.$
25. $D = \left\{ \begin{array}{l} (x, y) \\ \diagup -1 \leq x \leq 1, -2 \leq y \leq 0 \end{array} \right\},$
 $u(-1, y) = \psi_1(y) = y^2, \quad u(1, y) = \psi_2(y) = y^2 + 2y,$
 $u(x, -2) = \psi_3(x) = 2(x^2 - x), \quad u(x, 0) = \psi_4(x) = x^2 - 1,$
 $h_1 = h_2 = h = 0.2.$
26. $D = \left\{ \begin{array}{l} (x, y) \\ \diagup 2 \leq x \leq 4, -1 \leq y \leq 1 \end{array} \right\},$
 $u(2, y) = \psi_1(y) = 5 - 8y, \quad u(4, y) = \psi_2(y) = 18 - 7y,$
 $u(x, -1) = \psi_3(x) = 6x + 1, \quad u(x, 1) = \psi_4(x) = 7x - 17,$
 $h_1 = h_2 = h = 0.2.$
27. $D = \left\{ \begin{array}{l} (x, y) \\ \diagup 0 \leq x \leq 1, 0 \leq y \leq 1 \end{array} \right\},$
 $u(0, y) = \psi_1(y) = 1, \quad u(1, y) = \psi_2(y) = y + 1,$
 $u(x, 0) = \psi_3(x) = 1, \quad u(x, 1) = \psi_4(x) = x^2 + 1,$
 $h_1 = h_2 = h = 0.1.$
28. $D = \left\{ \begin{array}{l} (x, y) \\ \diagup 0 \leq x \leq 1, 0 \leq y \leq 1 \end{array} \right\},$
 $u(0, y) = \psi_1(y) = -5y^2 - 3y + 1, \quad u(1, y) = \psi_2(y) = -5y^2 - 15y + 7,$
 $u(x, 0) = \psi_3(x) = 4x^2 + 2x + 1, \quad u(x, 1) = \psi_4(x) = -4x^2 - 2x - 7,$
 $h_1 = h_2 = h = 0.1.$
29. $D = \left\{ \begin{array}{l} (x, y) \\ \diagup 0 \leq x \leq \frac{\pi}{2}, 0 \leq y \leq \frac{\pi}{2} \end{array} \right\},$
 $u(0, y) = \psi_1(y) = 10y^2 \left(\frac{\pi}{2} - y \right), \quad u\left(\frac{\pi}{2}, y\right) = \psi_2(y) = 0,$
 $u(x, 0) = \psi_3(x) = 15x \left(\frac{\pi^2}{4} - x^2 \right), \quad u\left(x, \frac{\pi}{2}\right) = \psi_4(x) = 30 \sin 2x,$

$$h_1 = h_2 = h = \frac{\pi}{20}.$$

30. $D = \left\{ \begin{array}{l} (x, y) \\ 0 \leq x \leq 1, 0 \leq y \leq 1 \end{array} \right\},$

$$u(0, y) = \psi_1(y) = 40y^2, \quad u(1, y) = \psi_2(y) = 40,$$

$$u(x, 0) = \psi_3(x) = 40 \sin \frac{\pi x}{2}, \quad u(x, 1) = \psi_4(x) = 40,$$

$$h_1 = h_2 = h = 0.1.$$

8.10. Лабораторная работа № 15. Решение однородного уравнения колебаний струны методом сеток по неявной схеме.

Уравнение гиперболического типа

$$\frac{\partial^2 u}{\partial t^2} - a_0^2 \frac{\partial^2 u}{\partial x^2} = f(x, t) \quad (8.10.1)$$

описывает поперечные колебания упругой натянутой струны. Его решение – функция $u(x, t)$ дает смещение участков струны в поперечном направлении.

Решим пример на это уравнение при условии $f(x, t) = 0$. Задача состоит в отыскании функции $u(x, t)$, удовлетворяющей в прямоугольнике $D = \left\{ \begin{array}{l} (x, t) \\ 0 \leq x \leq l, 0 \leq t \leq T \end{array} \right\}$ при $t > 0$ уравнению

$$\frac{\partial^2 u}{\partial t^2} = a_0^2 \frac{\partial^2 u}{\partial x^2}, \quad (8.10.2)$$

начальным условиям

$$u(x, 0) = \varphi_0(x), \quad \left. \frac{\partial u}{\partial t} \right|_{t=0} = \varphi_1(x), \quad 0 \leq x \leq l \quad (8.10.3)$$

и краевым условиям

$$u(0, t) = \psi_1(t), \quad u(l, t) = \psi_2(t), \quad 0 \leq t \leq T. \quad (8.10.4)$$

Для реализации разностной схемы решения задачи (8.10.2) – (8.10.4) построим в области D сетку $x_i = ih$, $i = 0, 1, \dots, n$, $l = hn$, $t_j = j\tau$, $j = 0, 1, \dots, m$, $T = tm$ и аппроксимируем производные от решения в узлах (x_i, t_j) , $i = 1, 2, \dots, n-1$, $j = 1, 2, \dots, m-1$ с помощью центральных разностных отношений (8.5.11) и (8.5.12) на пятиточечном шаблоне, изображенном на с. 206. Этот шаблон не требует выполнения какого-либо соотношения между шагами h и τ для обеспечения устойчивости решения.

Получим приближенное уравнение

$$\frac{u_{i,j+1} - 2u_{i,j} + u_{i,j-1}}{\tau^2} = a_0^2 \frac{u_{i+1,j+1} - 2u_{i,j+1} + u_{i-1,j+1}}{h^2}. \quad (8.10.5)$$

Погрешность замены дифференциального уравнения (8.10.2) разностным (8.10.5) составляет величину $O(h^2 + \tau^2)$. Полагая $\frac{h}{a_0 \tau} = \lambda$, получаем трехслойную разностную схему

$$u_{i-1,j+1} - (\lambda^2 + 2)u_{i,j+1} + u_{i+1,j+1} = \lambda^2(u_{i,j-1} - 2u_{i,j}). \quad (8.10.6)$$

Эта схема называется трехслойной потому, что связывает между собой приближенные значения $u_{i,j}$ функции $u(x, t)$ на трех временных слоях с номерами $j-1$, j и $j+1$.

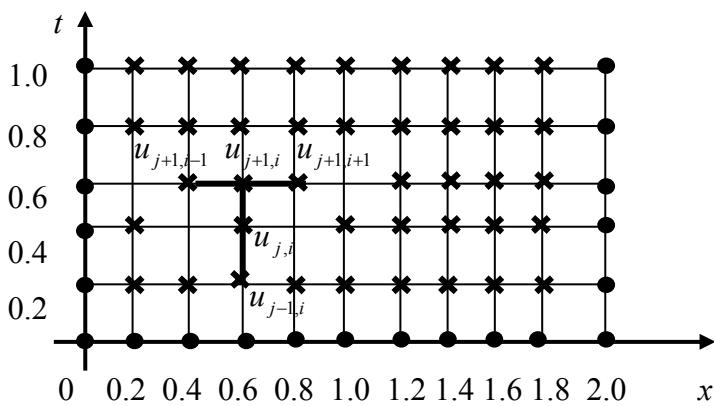
Система линейных алгебраических уравнений (8.10.6) обладает трехдиагональной матрицей и решается методом прогонки. Например, при $j=1$ система уравнений (8.10.6) приобретает вид

$$u_{i-1,2} + (\lambda^2 + 2)u_{i,2} + u_{i+1,2} = \lambda^2(u_{i,0} - 2u_{i,1}). \quad (8.10.7)$$

Значения $u_{i,0}$ известны из начальных условий (8.10.3), а для определения $u_{i,1}$ можно использовать один из возможных приемов, например,

$$u_{i,1} = u_{i,0} + \tau\varphi_1(x_i), \quad i = 0, 1, \dots, n. \quad (8.10.8)$$

Итак, найдем приближенные решения $u_{i,j}$ функции $u(x,t)$, удовлетворяющей уравнению $\frac{\partial^2 u}{\partial t^2} = \frac{\partial^2 u}{\partial x^2}$ с начальными условиями $u(x,0) = \varphi_0(x) = (x + 0.5)^2$, $u'_t(x,0) = (x + 1)\sin x$ и краевыми условиями $u(0,t) = \psi_1(t) = 0.5(0.5 + t)$, $u(2,t) = \psi_2(t) = 6.25$ в области $D = \{(x, t) / 0 \leq x \leq 2, 0 \leq t \leq 1\}$.



ми; помечен также используемый пятиточечный шаблон.

```

ORIGIN:=1
φ0(x):=(x + 0.5)^2 φ1(x):=(x + 1)*sin(x)
ψ1(t):=0.5*(0.5 + t) ψ2(t):=6.25
n:=11 m:=6 h:=0.2
i:=1..n x_i:=(i-1)*h u_{1,i}:=φ0(x_i) v_i:=φ1(x_i)
j:=1..m t_j:=(j-1)*h u_{j,1}:=ψ1(t_j) u_{j,n}:=ψ2(t_j)

```

Для вычисления решения на первом слое в данной лабораторной работе принят простейший способ, состоящий в том, что если положить

$$\frac{\partial u(x,0)}{\partial t} \approx \frac{u(x,\tau) - u(x,0)}{\tau}, \text{ то } u_{2,i} = u_{1,i} + \tau\varphi_1(x_i), \quad i = 2, 3, \dots, n-1, \quad (8.10.9)$$

то есть производная в дифференциальном уравнении первого порядка заменяется правым конечно-разностным отношением. Найдем решения на первом слое.

$$i := 2 \dots n-1 \quad u_{2,i} := u_{1,i} + h * v_i$$

Реализация неявной схемы вычисления решения исходного дифференциального уравнения во всех внутренних точках сетки начинается со второго слоя. Введем функции, вычисляющие трехдиагональную матрицу в левой части уравнений системы

$$u_{i-1,j+1} + 3u_{i,j+1} + u_{i+1,j+1} = u_{i,j-1} - 2u_{i,j}, \quad i = 2, 3, \dots, n-1, \quad j = 2, 3, \dots, m-1. \quad (8.10.10)$$

Систему уравнений (8.10.10) будем решать методом прогонки, разобранным в подразд. 5.4. Программа этого метода с небольшими изменениями уже использовалась нами в лабораторной работе № 6 (см. с. 89).

Для $j = 2, i = 2, 3, \dots, 10$ система (8.10.10) имеет следующий вид:

$$\begin{aligned}
i = 2: \quad & 3u_{2,3} + u_{3,3} = u_{2,1} - 2u_{2,2}, \\
i = 3: \quad & u_{2,3} + 3u_{3,3} + u_{4,3} = u_{3,1} - 2u_{3,2}, \\
& \dots \dots \dots
\end{aligned} \quad (8.10.11)$$

$$\begin{aligned} i=9: \quad & u_{8,3} + 3u_{9,3} + u_{10,3} = u_{9,1} - 2u_{9,2}, \\ i=10: \quad & u_{9,3} + 3u_{10,3} = u_{10,1} - 2u_{10,2}. \end{aligned}$$

Это трехдиагональная система с ярко выраженным диагональным преобладанием. Приведем текст подпрограммы, реализующий метод прогонки. Ее параметры: вектор \bar{b} содержит коэффициенты при неизвестных левой части системы (8.10.11), стоящие на главной диагонали, вектор \bar{a} - под главной диагональю, вектор \bar{c} - над главной диагональю. Вектор \bar{d} содержит свободные члены (правые части системы уравнений (8.10.11)). Подпрограмма выдает вектор \bar{x} - вектор решения системы. Для правильной работы программы необходимо задание ORIGIN:=1:

Функция `fun` вычисляет правую часть системы (8.10.10). Точки второго слоя находятся по формуле (8.10.9):

```

fun(i,j,u) := uj-1,i - 2 * uj,i
i := 1..n-2 bi := 3 ai := 1 ci := 1
a0 := 0 cn-1 := 0

```

Подпрограмма Setka составляет систему линейных уравнений (8.10.10) для каждого слоя сетки и решает ее методом прогонки: m - число слоев сетки по переменной t

	1
1	0
2	0.238
3	0.545
4	0.903
5	1.291
6	1.683
7	2.05
8	2.365
9	2.599
10	2.727
11	2.728

v =	b =	a =	c =
	3	0	1
	3	1	1
	3	1	1
	3	1	1
	3	1	1
	3	1	1
	3	1	1
	3	1	1
	3	1	0

```

Setka(a,b,c,m,u) := | n ← rows(b) + 2
                      | for j ∈ 2..m - 1
                      |   for i ∈ 2..n - 1
                      |     f_{i-1} ← fun(i,j,u)
                      |     xx ← progonka(a,b,c,f)
                      |     for i ∈ 2..n - 1
                      |       u_{j+1,i} ← xx_{i-1}
                      |
u

```

$$u1 := \text{Setka}(a, b, c, m, u)$$

	1	2	3	4	5	6	7	8	9	10	11
1	0.25	0.49	0.81	1.21	1.69	2.25	2.89	3.61	4.41	5.29	6.25
2	0.35	0.538	0.919	1.391	1.948	2.587	3.3	4.083	4.93	5.835	6.25
3	0.45	-0.13	-0.194	-0.315	-0.433	-0.594	-0.708	-0.991	-0.874	-1.835	6.25
4	0.55	0.189	0.233	0.421	0.525	0.817	0.798	1.505	0.752	2.918	6.25
5	0.65	-0.146	-0.069	-0.307	-0.166	-0.678	-0.029	-1.539	0.644	-2.772	6.25
6	0.75	0.191	-0.092	0.456	-0.24	1.121	-0.951	2.588	-2.23	3.564	6.25

Задание № 1. Найти приближенное решение однородного уравнения гиперболического типа $\frac{\partial^2 u}{\partial t^2} = \frac{\partial^2 u}{\partial x^2}$ в квадрате $D = \left\{ (x, t) \middle| 0 \leq x \leq 1, 0 \leq t \leq 1 \right\}$ с шагом $h = \tau = 0.1$, используя программы Setka и progonka .

$$1. \quad u(x,0) = \varphi_0(x) = x(x+1), \quad u'_t(x,0) = \cos x,$$

$$u(0,t) = \psi_1(t) = 0, \quad u(1,t) = \psi_2(t) = 2(t+1).$$

$$2. \quad u(x,0) = \varphi_0(x) = \cos \frac{\pi x}{2}, \quad u'_t(x,0) = x^2,$$

$$u(0,t) = \psi_1(t) = 1 + 2t, \quad u(1,t) = \psi_2(t) = 0.$$

$$3. \quad u(x,0) = \varphi_0(x) = x \sin \pi x, \quad u'_t(x,0) = (x+1)^2,$$

$$u(0,t) = \psi_1(t) = 2t, \quad u(1,t) = \psi_2(t) = 0.$$

4. $u(x,0) = \varphi_0(x) = x(2x - 0.5)$, $u'_t(x,0) = \cos 2x$,
 $u(0,t) = \psi_1(t) = t^2$, $u(1,t) = \psi_2(t) = 1.5$.

5. $u(x,0) = \varphi_0(x) = 2x(x+1) + 0.3$, $u'_t(x,0) = 2\sin x$,
 $u(0,t) = \psi_1(t) = 0.3$, $u(1,t) = \psi_2(t) = 4.3 + t$.

6. $u(x,0) = \varphi_0(x) = (1-x)\cos \frac{\pi x}{2}$, $u'_t(x,0) = 2x + 1$,
 $u(0,t) = \psi_1(t) = 2t + 1$, $u(1,t) = \psi_2(t) = 0$.

7. $u(x,0) = \varphi_0(x) = 0.5(x^2 + 1)$, $u'_t(x,0) = x \sin 2x$,
 $u(0,t) = \psi_1(t) = 0.5 + 3t$, $u(1,t) = \psi_2(t) = 1$.

8. $u(x,0) = \varphi_0(x) = x^2 \cos \pi x$, $u'_t(x,0) = x^2(x+1)$,
 $u(0,t) = \psi_1(t) = 0.5t$, $u(1,t) = \psi_2(t) = t - 1$.

9. $u(x,0) = \varphi_0(x) = \sin \pi x \cdot \cos x$, $u'_t(x,0) = \left(\frac{1}{2}\right)^x$,
 $u(0,t) = \psi_1(t) = 0$, $u(1,t) = \psi_2(t) = 0$.

10. $u(x,0) = \varphi_0(x) = \begin{cases} 2x, & 0 \leq x \leq 0.6, \\ -2.5x + 2.5, & 0.6 < x \leq 1 \end{cases}$, $u'_t(x,0) = \begin{cases} 0, & 0 \leq x \leq 0.6, \\ 1, & 0.6 < x \leq 1 \end{cases}$,
 $u(0,t) = \psi_1(t) = t^2$, $u(1,t) = \psi_2(t) = e^t - 1$.

11. $u(x,0) = \varphi_0(x) = 4x^2(x-1)$, $u'_t(x,0) = \sin \frac{\pi x}{2}$,
 $u(0,t) = \psi_1(t) = 0$, $u(1,t) = \psi_2(t) = 0$.

12. $u(x,0) = \varphi_0(x) = \sin xe^x \cdot \sin(x-1) + 1$, $u'_t(x,0) = e^{-x}$,
 $u(0,t) = \psi_1(t) = 1$, $u(1,t) = \psi_2(t) = 1$.

13. $u(x,0) = \varphi_0(x) = \begin{cases} x, & 0 \leq x \leq 0.8, \\ -x, & 0.8 < x \leq 1 \end{cases}$, $u'_t(x,0) = e^{\frac{-x}{2}}$,
 $u(0,t) = \psi_1(t) = \cos \pi t - 1$, $u(1,t) = \psi_2(t) = t^2 + 0.6$.

14. $u(x,0) = \varphi_0(x) = x \sin(2(x-1))$, $u'_t(x,0) = 3x - 1$,
 $u(0,t) = \psi_1(t) = 0$, $u(1,t) = \psi_2(t) = 0$.

15. $u(x,0) = \varphi_0(x) = 1.2x - x^2$, $u'_t(x,0) = (x + 0.6)\sin x$,
 $u(0,t) = \psi_1(t) = 0$, $u(1,t) = \psi_2(t) = 0.2 + 0.5t$.

16. $u(x,0) = \varphi_0(x) = (1-x^2)\cos \pi x$, $u'_t(x,0) = 2x + 0.6$,
 $u(0,t) = \psi_1(t) = 1 + 0.4t$, $u(1,t) = \psi_2(t) = e^t - 1$.

17. $u(x,0) = \varphi_0(x) = x \cos \pi x$, $u'_t(x,0) = x(2-x)$,
 $u(0,t) = \psi_1(t) = 2t$, $u(1,t) = \psi_2(t) = -1$.

18. $u(x,0) = \varphi_0(x) = (x + 0.5)(x-1)$, $u'_t(x,0) = \sin(x + 0.2)$,
 $u(0,t) = \psi_1(t) = t - 0.5$, $u(1,t) = \psi_2(t) = 3t$.

19. $u(x,0) = \varphi_0(x) = (x + 0.2)\sin \frac{\pi x}{2}$, $u'_t(x,0) = 1 + x^2$,
 $u(0,t) = \psi_1(t) = 0$, $u(1,t) = \psi_2(t) = 1.2(t+1)$.

20. $u(x,0) = \varphi_0(x) = 0.5x(x+1)$, $u'_t(x,0) = x \cos x$,
 $u(0,t) = \psi_1(t) = 2t^2$, $u(1,t) = \psi_2(t) = 1$.

$$21. \quad u(x,0) = \varphi_0(x) = (x+2)(0.5x+1), \quad u'_t(x,0) = 2\cos\left(x + \frac{\pi}{6}\right), \\ u(0,t) = \psi_1(t) = 2, \quad u(1,t) = \psi_2(t) = 4.5 - 3t.$$

$$22. \quad u(x,0) = \varphi_0(x) = 0.4(x+0.5)^2, \quad u'_t(x,0) = x\sin(x+0.6), \\ u(0,t) = \psi_1(t) = 0.1 + 0.5t, \quad u(1,t) = \psi_2(t) = 0.9.$$

$$23. \quad u(x,0) = \varphi_0(x) = \begin{cases} x, & 0 \leq x \leq 0.5, \\ 0.5, & 0.5 < x \leq 1 \end{cases}, \quad u'_t(x,0) = 2x, \\ u(0,t) = \psi_1(t) = \ln\left(1 - \frac{x^2}{2}\right), \quad u(1,t) = \psi_2(t) = 0.5 + t^3.$$

$$24. \quad u(x,0) = \varphi_0(x) = \begin{cases} 0, & 0 \leq x \leq 0.3, \\ 2, & 0.3 < x \leq 1 \end{cases}, \quad u'_t(x,0) = e^{-2x}, \\ u(0,t) = \psi_1(t) = \sin t, \quad u(1,t) = \psi_2(t) = 2 - \ln(1+t).$$

$$25. \quad u(x,0) = \varphi_0(x) = 10x(x^3 - 1), \quad u'_t(x,0) = \cos\frac{\pi x}{2}, \\ u(0,t) = \psi_1(t) = 0, \quad u(1,t) = \psi_2(t) = 0.$$

$$26. \quad u(x,0) = \varphi_0(x) = (x-1)\ln(1+x), \quad u'_t(x,0) = \frac{1}{\frac{t}{t+1}}, \\ u(0,t) = \psi_1(t) = 0, \quad u(1,t) = \psi_2(t) = 0.$$

$$27. \quad u(x,0) = \varphi_0(x) = (x+0.5)(x+1), \quad u'_t(x,0) = \cos(x+0.3), \\ u(0,t) = \psi_1(t) = 0.5, \quad u(1,t) = \psi_2(t) = 3 - 2t.$$

$$28. \quad u(x,0) = \varphi_0(x) = (x+0.4)\sin\pi x, \quad u'_t(x,0) = (x+1)^2, \\ u(0,t) = \psi_1(t) = 0.5t, \quad u(1,t) = \psi_2(t) = 0.$$

$$29. \quad u(x,0) = \varphi_0(x) = (x+0.4)\cos\frac{\pi x}{2}, \quad u'_t(x,0) = 0.3(x^2 + 1), \\ u(0,t) = \psi_1(t) = 0.4, \quad u(1,t) = \psi_2(t) = 1.2t.$$

$$30. \quad u(x,0) = \varphi_0(x) = 0.5(x+1)^2, \quad u'_t(x,0) = (x+0.5)\cos\pi x, \\ u(0,t) = \psi_1(t) = 0.5, \quad u(1,t) = \psi_2(t) = 2 - 3t.$$

Библиографический список.

1. Амосов А.А., Дубинский Ю.А., Копченова Н.В. Вычислительные методы для инженеров. М.: Высшая школа, 1994.
2. Бахвалов Н.С., Жидков Н.П., Кобельков Г.М. Численные методы. М.: Наука, 1987.
3. Бахвалов Н.С., Лапин А.В., Чижонков Е.В. Численные методы в задачах и упражнениях. М.: Высшая школа, 2000.
4. Волков Е.А. Численные методы. М.: Наука, 1987.
5. Воробьев Г.Н., Данилова А.Н. Практикум по вычислительной математике. М.: Высшая школа, 1990.
6. Гантмахер Ф.Р. Теория матриц. М.: Наука, 1988.
7. Демидович Б.П., Марон И.А., Шувалова Э.З. Численные методы анализа. М.:ГИФМЛ, 1963.
8. Демидович Б.П., Марон И.А. Основы вычислительной математики. М.: Наука, 1970.
9. Дэннис Дж., Шнабель Р. Численные методы безусловной оптимизации и решения нелинейных уравнений. М.: Мир, 1988.

10. *Mathcad 6.0 Plus*. Финансовые, инженерные и научные расчеты в среде Windows 95. М.: Филинъ, 1996.
11. *Очков В.Ф.* Mathcad 8 Pro для студентов и инженеров. М.: Компьютерпресс, 1999.
12. *Плис А.И., Сливина Н.А.* Лабораторный практикум по высшей математике. М.: Высшая школа, 1994.
13. *Плис А.И., Сливина Н.А.* Mathcad: математический практикум. М.: Финансы и статистика, 1999.
14. *Тихонов А.Н., Самарский А.А.* Уравнения математической физики. М.: Наука, 1966.
15. *Химмельблау Д.* Прикладное нелинейное программирование. М.: Мир, 1975.
16. *Форсайт Дж., Малькольм М., Моулер К.* Машины методы математических вычислений. М.: Мир, 1980.

Оглавление

1. Элементарная теория погрешностей; вычислительные задачи, методы и алгоритмы	Ошибка! Закладка не определена.
1.1. Источники и классификация погрешностей результата численного эксперимента	Ошибка! Закладка не определена.
1.2. Погрешности чисел	Ошибка! Закладка не определена.
1.3. Погрешности арифметических операций	Ошибка! Закладка не определена.
1.4. Погрешности функций	Ошибка! Закладка не определена.
1.5. Особенности машинной арифметики	Ошибка! Закладка не определена.
1.6. Лабораторная работа № 1. Определение абсолютной и относительной погрешностей приближенных чисел. Оценка погрешностей результата	Ошибка! Закладка не определена.
1.7. Корректность вычислительной задачи	Ошибка! Закладка не определена.
1.8. Обусловленность вычислительной задачи	Ошибка! Закладка не определена.
1.9. Вычислительные методы, их классификация	Ошибка! Закладка не определена.
2. Приближение функций	25
2.1. Задача приближения функций	Ошибка! Закладка не определена.
2.2. Интерполяция обобщенными многочленами	Ошибка! Закладка не определена.
2.3. Полиномиальная интерполяция. Многочлен Лагранжа	Ошибка! Закладка не определена.
2.4. Погрешность интерполяции	Ошибка! Закладка не определена.
2.5. Конечные разности и их свойства	Ошибка! Закладка не определена.
2.6. Разделенные разности и их свойства	Ошибка! Закладка не определена.
2.7. Интерполяционный многочлен Ньютона	Ошибка! Закладка не определена.
2.8. Вычислительная схема Эйткена	Ошибка! Закладка не определена.
2.9. Лабораторная работа № 2. Интерполирование и экстраполирование данных. Интерполяционный многочлен Лагранжа	Ошибка! Закладка не определена.
2.10. Интерполяционный многочлен Ньютона с конечными разностями	Ошибка! Закладка не определена.
2.11. Лабораторная работа № 3. Интерполирование и экстраполирование данных. Интерполяционный многочлен Ньютона	Ошибка! Закладка не определена.
2.12. Интерполяционные формулы Гаусса, Стирлинга и Бесселя	Ошибка! Закладка не определена.
3. Метод наименьших квадратов и специальные интерполяционные многочлены	Ошибка! Закладка не определена.
3.1. Постановка задачи и вывод формул метода наименьших квадратов	Ошибка! Закладка не определена.
3.2. Лабораторная работа № 4. Аппроксимация функции по методу наименьших квадратов	Ошибка! Закладка не определена.
3.3. Глобальная полиномиальная интерполяция	Ошибка! Закладка не определена.
3.4. Чувствительность интерполяционного многочлена к погрешностям входных данных	Ошибка! Закладка не определена.
3.5. Многочлены Чебышева	Ошибка! Закладка не определена.
3.6. Решение задачи минимизации оценки погрешности	Ошибка! Закладка не определена.
3.7. Ряд Фурье по многочленам Чебышева	Ошибка! Закладка не определена.
3.8. Лабораторная работа № 5. Экономизация степенных рядов	Ошибка! Закладка не определена.
3.9. Локальная интерполяция	Ошибка! Закладка не определена.
3.10. Сплайны, их свойства и построение	Ошибка! Закладка не определена.
3.11. Погрешность приближения кубическими сплайнами	Ошибка! Закладка не определена.
3.12. Лабораторная работа № 6. Аппроксимация функций кубическими сплайнами	Ошибка! Закладка не определена.
3.13. Тригонометрическая интерполяция. Дискретное преобразование Фурье и его реализация на ЭВМ	Ошибка! Закладка не определена.
3.14. Матричная форма записи дискретного преобразования Фурье (ДПФ)	Ошибка! Закладка не определена.
3.15. Алгоритм реализации ДПФ	Ошибка! Закладка не определена.
3.16. Пример реализации алгоритма ДПФ при $2N = 4$	Ошибка! Закладка не определена.
3.17. Лабораторная работа № 7. Дискретное преобразование Фурье	Ошибка! Закладка не определена.
4. Численное дифференцирование и численное интегрирование	Ошибка! Закладка не определена.
4.1. Простейшие формулы численного дифференцирования для первой производной	Ошибка! Закладка не определена.
4.2. Формулы численного дифференцирования для второй производной	Ошибка! Закладка не определена.
4.3. Формулы численного дифференцирования, основанные на интерполяции алгебраическими многочленами	Ошибка! Закладка не определена.
4.4. Обусловленность формул численного дифференцирования	Ошибка! Закладка не определена.
4.5. Простейшие квадратурные методы численного интегрирования	Ошибка! Закладка не определена.
4.6. Оценка погрешностей простейших квадратурных формул	Ошибка! Закладка не определена.
4.7. Квадратурные формулы интерполяционного типа	Ошибка! Закладка не определена.
4.8. Квадратурные формулы Гаусса	Ошибка! Закладка не определена.
4.9. Лабораторная работа № 8. Численное дифференцирование и численное интегрирование функций	Ошибка! Закладка не определена.
5. Решение систем линейных алгебраических уравнений и проблемы собственных значений	Ошибка! Закладка не определена.
5.1. Нормы векторов и матриц и их свойства	Ошибка! Закладка не определена.
5.2. Обусловленность задачи решения системы линейных алгебраических уравнений	Ошибка! Закладка не определена.
5.3. Метод Гаусса (схема единственного деления)	Ошибка! Закладка не определена.
5.4. Метод прогонки	Ошибка! Закладка не определена.
5.5. Метод простых итераций	Ошибка! Закладка не определена.
5.6. Сходимость метода простых итераций	Ошибка! Закладка не определена.

5.7. Метод Зейделя	Ошибка! Закладка не определена.
5.8. Сходимость метода Зейделя.....	Ошибка! Закладка не определена.
5.9. Лабораторная работа № 9. Решение систем линейных алгебраических уравнений методом простых итераций	Ошибка! Закладка не определена.
5.10. Постановка задачи нахождения собственных чисел.....	Ошибка! Закладка не определена.
5.11. Подобные матрицы	Ошибка! Закладка не определена.
5.12. Локализация собственных значений	Ошибка! Закладка не определена.
5.13. Степенной метод.....	Ошибка! Закладка не определена.
5.14. Вычисление собственных векторов методом обратных итераций	Ошибка! Закладка не определена.
5.15. Лабораторная работа № 10. Вычисление собственных значений (чисел) и векторов матриц	Ошибка! Закладка не определена.
6. Численные методы решения нелинейных уравнений и систем	Ошибка! Закладка не определена.
6.1. Решение нелинейных уравнений	Ошибка! Закладка не определена.
6.2. Метод Ньютона для уравнений	Ошибка! Закладка не определена.
6.3. Сходимость метода Ньютона и трудности его применения.....	Ошибка! Закладка не определена.
6.4. Метод Ньютона решения систем нелинейных уравнений	Ошибка! Закладка не определена.
6.5 Сходимость метода Ньютона для систем нелинейных уравнений	155
6.6. Модификации метода Ньютона	Ошибка! Закладка не определена.
6.7. Лабораторная работа № 11. Решение систем нелинейных уравнений методом Ньютона	Ошибка! Закладка не определена.
7. Численные методы решения задачи Коши для обыкновенных дифференциальных уравнений и систем дифференциальных уравнений	Ошибка! Закладка не определена.
7.1. Задача Коши для дифференциального уравнения первого порядка	Ошибка! Закладка не определена.
7.2. Численные методы решения задачи Коши. Основные понятия и определения	Ошибка! Закладка не определена.
7.3. Решение с помощью рядов Тейлора.....	Ошибка! Закладка не определена.
7.4. Методы Рунге - Кутты.....	Ошибка! Закладка не определена.
7.5. Анализ ошибок, возникающих при использовании методов Рунге - Кутты	Ошибка! Закладка не определена.
7.6. Методы прогноза и коррекции.....	Ошибка! Закладка не определена.
7.7. Сравнение методов	Ошибка! Закладка не определена.
7.8. Лабораторная работа № 12. Методы интегрирования обыкновенных дифференциальных уравнений	Ошибка! Закладка не определена.
7.9. Решение задачи Коши для систем обыкновенных дифференциальных уравнений	Ошибка! Закладка не определена.
7.10. Сведение задачи Коши для уравнения n -го порядка к задаче Коши для системы уравнений первого порядка	Ошибка! Закладка не определена.
7.11. Лабораторная работа № 13. Численное интегрирование систем дифференциальных уравнений первого порядка	Ошибка! Закладка не определена.
8. Дифференциальные уравнения с частными производными (уравнения математической физики)	Ошибка! Закладка не определена.
8.1. Классификация уравнений математической физики	Ошибка! Закладка не определена.
8.2. Простейшие задачи, приводящие к дифференциальным уравнениям в частных производных	Ошибка! Закладка не определена.
8.3. Границные и начальные условия дифференциальных уравнений с частными производными	Ошибка! Закладка не определена.
8.4. Уравнения параболического типа. Явные и неявные схемы.....	Ошибка! Закладка не определена.
8.5. Уравнения гиперболического типа.....	Ошибка! Закладка не определена.
8.6. Уравнения эллиптического типа	Ошибка! Закладка не определена.
8.7. Свойства разностных схем для дифференциальных уравнений: способность аппроксимировать исходную дифференциальную задачу, устойчивость и сходимость...	Ошибка! Закладка не определена.
8.8. Некоторые обобщения.....	Ошибка! Закладка не определена.
8.9. Лабораторная работа № 14. Решение задачи Дирихле для уравнения Лапласа методом сеток	Ошибка! Закладка не определена.
8.10. Лабораторная работа № 15. Решение однородного уравнения колебаний струны методом сеток по неявной схеме.	Ошибка! Закладка не определена.
<i>Библиографический список</i>	228

Шапорев Сергей Дмитриевич

Методы вычислительной математики и их приложения

Редактор Г.М. Звягина
Корректор А.А. Баутдинова

Подписано в печать 30. 05. 2002. Формат 60x84/16. Бумага документная. Печать трафаретная. Усл. - печ. л. 14,25.
Уч. - изд. л. 20. Тираж 130 экз. Заказ №

Балтийский государственный технический университет
Типография БГТУ
198005, С.-Петербург, 1-я Красноармейская ул., д.1