

Министерство образования и науки РФ

Филиал ФГБОУ ВПО «ЮУрГУ» (НИУ)

В г. Нижневартовске

Кафедра «Информатика»

Н.Р. ЖАРОВА

ЧИСЛЕННЫЕ МЕТОДЫ В ИНЖЕНЕРНЫХ РАСЧЕТАХ

УЧЕБНОЕ ПОСОБИЕ

НИЖНЕВАРТОВСК

2013

1

УДК

ББК

Ж

Рецензенты:

Доктор педагогических наук, профессор, заведующий кафедрой «Информационная безопасность» ФГБОУ ВПО «Сибирская автомобильно-дорожная академия» (СибАДИ)

З.В. Семенова;

Кандидат физико-математических наук, доцент кафедры «Информатика» ФГБОУ ВПО «Южно-Уральский государственный университет» Н.В. Абрамов

Н.Р. Жарова

Численные методы в инженерных расчетах: Учебное пособие.- Нижневартовск: Изд-во Нижневарт. Гуман. Ун-та, 2013.- 160 с.

В пособии рассмотрены численные методы решения нелинейных уравнений; представлены точные и приближенные методы решения систем алгебраических уравнений. В главах 4 и 5 представлены методы интерполяции и аппроксимации функций. Главы 6 и 7 содержат сведения и приближенных вычислениях интегралов и дифференциальных уравнениях. В каждой главе имеются вопросы для самоконтроля и множество примеров решения задач на ПК при помощи универсальной математической системы MathCAD. Каждая глава содержит задания к лабораторной работе и 15 вариантов заданий к ней, приведены примеры, иллюстрирующие выполнение лабораторной работы.

Данное пособие соответствует ФГОС ВПО для направлений обучения 200100.62. – Приборостроение, 230100.62- Информатика и вычислительная техника, 231000.62 - Программная инженерия и других технических и направлений и специальностей. Пособие может быть полезно при изучении и преподавании дисциплин математического и естественнонаучного циклов.

©Жарова Н.Р., 2013

©Издательство НвГУ, 2013

ОГЛАВЛЕНИЕ

Введение	5
Глава 1. Элементы теории погрешностей.....	7
1.1 Источники погрешностей и приближений.....	7
1.2. Приближенные числа.....	9
1.3. Прямая задача теории погрешностей.....	11
1.4. Обратная задача теории погрешностей.....	14
1.5. Особенности машинной арифметики.....	14
Глава 2. Решение нелинейных уравнений.....	19
2.1 Понятия и определения.....	19
2.2 Локализация (отделение) корней уравнения.....	21
2.3 Уточнение корней уравнения.....	23
2.4 Методы уточнения корней.....	
2.4.1. Метод половинного деления (бисекции, дихотомии)	
2.4.2 Метод хорд.....	
2.4.3. Метод Ньютона (метод касательных) Лабораторная работа 1	
Глава 3. Решение систем линейных уравнений	
3.1 Понятия и определения. Прямые методы решения СЛАУ	
3.2 Решение систем линейных уравнений матричным способом	
3.3 Решение систем линейных уравнений методом Крамера	
3.4 Решение систем линейных уравнений методом Гаусса Лабораторная работа 2	
3.5 Приближенные методы решения систем линейных уравнений	
3.5.1 Решение систем линейных уравнений методом итераций	
3.5.2 Решение систем линейных уравнений методом Зейделя Лабораторная работа № 3	
Глава 4. Интерполирование функций	
4.1 Интерполяция	
4.1.1 Интерполяционный многочлен Лагранжа	
4.1.2 Интерполяционный многочлен Ньютона	

Лабораторная работа № 4

4.1.3 Сплайн-интерполяция

Глава 5. Аппроксимация функций

5.1. Метод наименьших квадратов

5.1.1 Линейная регрессия

5.1.2 Квадратичная регрессия

5.1.3 Геометрическая регрессия

5.1.4 Тригонометрическая функция

5.1.5 Логарифмическая функция

5.1.6 Дробно-линейная функция

Лабораторная работа № 5

Глава 6. Численное интегрирование

6.1 Интерполяционные методы Ньютона-Котеса

6.1.1 Методы прямоугольников

6.1.2 Метод трапеций

6.1.3 Метод Симпсона (метод парабол)

6.1.4 Погрешность формул Ньютона-Котеса

6.1.5 Вычисление интеграла с заданной точностью

Лабораторная работа №6

Глава 7. Численные методы решения обыкновенных дифференциальных уравнений

7.1 Интегрирование дифференциальных уравнений с помощью степенных рядов

7.2. Метод последовательных приближений

7.3. Метод Эйлера

7.4. Модификации метода Эйлера

7.5. Метод Рунге-Кутты

Лабораторная работа №6

Библиографический список

ВВЕДЕНИЕ

Численные методы в инженерных расчетах — это наука о методах решения вычислительных задач на компьютере. Она появилась от необходимости решать практические задачи, такие, как управление сложными технологическими процессами, управление полётом ракет, моделирование физических процессов (процесса ядерного распада, химических реакций, роста кристаллов и др.).

Задачами вычислительной математики занимались такие выдающиеся учёные, как Эйлер, Лагранж, Чебышёв, Якоби, Лежандр, фон Нейман и многие другие. Они, часто занимаясь сложными вычислениями вручную на бумаге, невольно заложили основы науки об эффективных безошибочных вычислениях на компьютерах. Всем известно, что компьютеры имеют дело с числами с ограниченным количеством знаков после запятой. Однако именно эти «мелочи» могут сильно исказить результаты численных расчётов. Появился важнейший раздел вычислительной математики — теория устойчивости вычислительных методов.

Данное учебное пособие написано в соответствии с ФГОС ВПО третьего поколения, учебными планами по направлению обучения 200100.62. - Приборостроение. Стандарты предыдущих поколений прописывали конкретные разделы дисциплины «Вычислительная математика», знаниями которых должны обладать будущие выпускники:

- особенности математических вычислений, реализуемых на ЭВМ;
- теоретические основы численных методов: погрешности вычислений;
- устойчивость и сложность алгоритма (по памяти по времени);
- численные методы линейной алгебры;
- решение нелинейных уравнений и систем;
- численное интегрирование и дифференцирование;
- решение обыкновенных дифференциальных уравнений;
- методы приближения и аппроксимации функций;
- равномерное приближение функций;

- математические программные системы.

ФГОС ВПО третьего поколения предусматривают проектирование компетентностно-ориентированных рабочих программ дисциплин, входящих в учебный план.

Изучение вышеперечисленных разделов данной дисциплины, предусматривает формирование следующих компетентностей бакалавра:

общекультурной:

- использует основные законы естественнонаучных дисциплин в профессиональной деятельности, применяет методы математического анализа и моделирования, теоретического и экспериментального исследования (ОК-10);

профессиональной (ПК):

- освоение методик использования программных средств для решения практических задач (ПК-2).

Дисциплина не входит в перечень базовых дисциплин ФГОС ВПО, но является составляющей вариативной части базовых дисциплин естественнонаучного цикла, определенных ООП вуза.

Пособие содержит необходимые теоретические сведения, примеры, иллюстрирующие применение того или иного вычислительного метода, пример выполнения задания в среде универсальной математической системы MathCAD, задания для самостоятельного решения. Пособие предназначено для студентов, обучающихся по техническим, экономическим и другим направлениям, в составе основных образовательных программ которых формируются указанные компетентности, может быть полезно аспирантам и преподавателям.

Каждая глава пособия содержит 15 вариантов заданий для лабораторной работы, а также перечень вопросов для защиты выполненной работы. Пособие предназначено для студентов очной и заочной форм обучения.

ГЛАВА 1

ЭЛЕМЕНТЫ ТЕОРИИ ПОГРЕШНОСТЕЙ

1.1 Источники погрешностей и приближений

Процесс решения задачи с использованием вычислительной техники проходит целый ряд этапов: построение математической модели, разработка или выбор численного метода, разработка алгоритма, программирование, проведение вычислений. Некоторые из этих этапов могут являться источником погрешностей и оказывать тем самым свое влияние на достоверность (точность) окончательного результата. Рассмотрим основные источники погрешностей:

- исходные данные задачи, полученные экспериментально или в ходе расчетов, часто являются основным источником погрешностей;
- используемая при решении задачи математическая модель также вносит погрешность в получаемый результат в виду того, что она является лишь приближенным описанием реального процесса или явления;
- погрешности исходных данных и погрешность математической модели относятся к виду *неустранимых погрешностей* в том смысле, что в ходе последующих вычислений их нельзя устранить.

Если для решения математической задачи используется приближенный (например, численный) метод, то еще не приступив к вычислениям, мы допускаем новую погрешность, называемую *погрешностью метода*.

Погрешность численного метода теоретически может быть уменьшена до любого значения. Однако на практике ограничиваются тем, чтобы довести погрешность метода до величины, в несколько раз меньшей неустранимой погрешности. Дальнейшее повышение точности метода не приведет к повышению точности окончательного результата, а лишь увеличит стоимость расчетов из-за увеличения объема вычислений.

При вычислениях с помощью компьютера неизбежны *погрешности округлений*, связанные с ограниченностью разрядной сетки вычислительной машины. Несмотря на то, что при решении сложных задач выполняются миллиарды и триллионы операций, это вовсе не означает механического умножения погрешности при одном округлении на число операций, так как при отдельных действиях погрешности могут компенсировать друг друга. Вместе с тем иногда погрешности округлений в сочетании с плохо организованным алгоритмом могут сильно исказить результаты или даже привести к абсурдным результатам.

Погрешность является мерой точности результата. Для количественной характеристики этой меры используют понятия *абсолютной и относительной погрешностей*.

Определение. Пусть x – точное и неизвестное значение какой-либо величины, a – ее известное приближенное значение. Тогда абсолютную величину разности между соответствующим точным числом x и его приближенным значением называют абсолютной погрешностью приближенного числа

$$\Delta = |x - x^*|. \quad (1)$$

В качестве абсолютной погрешности результатов измерений часто принимают половину цены деления измерительного прибора или среднеквадратичное отклонение (при проведении серии измерений).

Определение. Предельной абсолютной погрешностью приближенной величины x^* называется величин

$$\Delta_{x^*} \geq |x - x^*| \quad (2)$$

(не меньшая абсолютной погрешности этого числа).

Предельную абсолютную погрешность часто называют просто абсолютной погрешностью. Иногда записывают $x = x^* + \Delta_{x^*}$.

Например, запись $x = 2,123 \pm 0,003$ или $x = 2,123 \pm 3 \cdot 10^{-3}$ означает, что $2,123 - 0,003 \leq x \leq 2,123 + 0,003$.

Определение. Предельной относительной погрешностью или просто относительной погрешностью δ приближенной величины

x^* называют отношение предельной абсолютной погрешности Δ_x^* к приближенной величине x^* :

$$\delta = \left| \frac{\Delta_x^*}{x^*} \right| \quad (3)$$

Относительную погрешность часто выражают в процентах.

1.2 Приближенные числа

При работе с приближенными числами используют понятия *значащих и верных значащих цифр*.

Определение. Значащими цифрами приближенного числа x^* называются все цифры в его записи, начиная с первой ненулевой слева.

Например, 0.001234 – четыре значащие цифры, 1.2345- пять значащих цифр, 4.00567 - шесть значащих цифр. (Все значащие цифры подчеркнуты).

Определение. Значащая цифра приближенного числа x^* называется *верной*, если абсолютная погрешность числа не превосходит половины единицы разряда, в котором находится это число. Значащие цифры разрядов, где не выполняется это условие, называются *сомнительными*.

Пример.

Пусть для приближенного числа $x^* = 72,356$ известна абсолютная погрешность $\Delta_x^* = 0.04$. Требуется определить его верные значащие цифры.

Проверим цифру 7. Половина единицы ее разряда $\frac{10}{2} = 5 \geq 0.04$. Значит, она верная. Для цифры 2: $\frac{1}{2} = 0.5 \geq 0.04$ -

тоже верная. Верной будет и цифра 3: $\frac{0.1}{2} = 0.05 \geq 0.04$;

для цифры 5 имеем: $\frac{0.01}{2} = 0.005 < 0.04 \Rightarrow$ сомнительная

цифра, Для цифры 6: $\frac{0.001}{2} = 0.0005 < 0.04 \Rightarrow$ сомнительная цифра.

Количество верных значащих цифр тесно связано с величиной относительной погрешности числа. В частности, если приближенное число x^* содержит N верных значащих цифр, то для относительной погрешности имеет место соотношение $\delta \approx 10^{-N}$. Это позволяет легко оценивать точность приближенного значения. Так, например, если дано число $x^* = 2.345$ и сказано, что в его записи оставлены только верные цифры, то относительная погрешность этого числа $\delta \approx 10^{-4}$.

При изменении формы записи приближенных чисел (например, при записи в форме с плавающей точкой) количество *значащих цифр не должно меняться*, т.е. нужно соблюдать равносильность преобразований. Например, записи $2400 = 0.2400 \cdot 10^4$ и $0.710 \cdot 10^2 = 71,0$ *равносильные*, а записи $2400 = 0.24 \cdot 10^4$

$0.710 \cdot 10^2 = 71$ *неравносильные*.

Абсолютную и относительную погрешности обычно записывают в виде числа, содержащего одну или две значащие цифры. При округлении погрешностей округление всегда производится в большую сторону.

Информацию о том, что число x^* является приближенным значением числа x с абсолютной погрешностью Δ_{x^*} часто записывают в виде $x = x^* \pm \Delta_{x^*}$, причем числа x^* и Δ_{x^*} принято записывать с одинаковым числом знаков после запятой.

Например, если в ходе вычислений, которые выполнялись с сохранением восьми десятичных знаков, было получено приближенное число $x^* = 12.46874967$ и известно при этом, что абсолютная погрешность этого числа $\Delta_{x^*} = 0.02$, то результат вычислений следует записать в виде $x \approx 12.46 \pm 0.02$ или $x \approx 12.46 \pm 2 \cdot 10^{-2}$.

Если же в результате измерений (или вычислений) получено число $x^* = 10428$ с абсолютной погрешностью $\Delta_{x^*} = 23$, то этот результат следует записать в виде

$$x \approx (104.2 \pm 0.3) \cdot 10^2.$$

Информацию о том, что число x^* является приближенным значением числа x с относительной погрешностью δ записывают в виде $x \approx x^*(1 \pm \delta)$. Например, записи $x \approx 1.123(1 \pm 0.003)$, $x \approx 1.123(1 \pm 0.3\%)$ означают, что $(1 - 0.003) \cdot 1.123 \leq x \leq (1 + 0.003) \cdot 1.123$.

1.3 Прямая задача теории погрешностей

При вычислениях с приближенными числами важной задачей является оценка степени влияния погрешностей исходных данных на точность окончательного результата. Это необходимо не только для правильного учета вычислительных погрешностей, но также для определения возможных путей их уменьшения.

В частности, при вычислении значений функций, аргументами которых являются приближенные числа, возникает вопрос о погрешности вычисляемых значений. Определение величины погрешности результата по известным погрешностям исходных данных составляет *прямую задачу теории погрешностей*.

Пусть $y = f(x)$ – функция одного неизвестного. Аргумент функции задан приближенным числом x^* с абсолютной погрешностью Δ_{x^*} . Абсолютная погрешность значения функции согласно (2) $\Delta_{x^*} \geq |x - x^*|$ определяется так

$$\Delta_{y^*} \geq |y - y^*|.$$

Рассматривая разность точного (неизвестного) и приближенного значений функции как приращение функции, вызванное приращением аргумента Δ_{x^*} , и применяя *теорему Лагранжа* о

замене приращения функции ее дифференциалом, можно получить следующее выражение для Δ_y^* :

$$\Delta_y^* = \Delta_x^* \left| f'(x^*) \right| \quad (4)$$

При условии дифференцируемости функции $y = f(x)$.

Эта формула обобщается на случай функции многих переменных

$$y = f(x_1, x_2, \dots, x_n) : \Delta_y^* = \sum_{i=1}^n \Delta_{x_i}^* \left| \frac{\partial}{\partial x_i} f(x_1, x_2, \dots, x_n) \right| \quad (5).$$

Примеры.

$$1. \quad y = \ln x, \Delta_y^* = \Delta_x^* \left| \frac{1}{x} \right|.$$

$$2. \quad z(x, y) = 2x^3 + x \sin y, \Delta_z^* = \Delta_x^* |6x^2 + \sin y| + \Delta_y^* |x \cos y|.$$

Используя формулу (5) можно легко получить формулы для определения абсолютных и относительных погрешностей результатов арифметических операций. Так, например, для суммы или разности двух слагаемых a и b можно записать функцию двух переменных $f(a, b) = a \pm b$, абсолютная погрешность которой согласно (5) определяется как

$$\Delta(a^* \pm b^*) = \Delta_a^* |f'_a| \pm \Delta_b^* |f'_b| = \Delta_a^* \pm \Delta_b^* \quad (6)$$

➤ **З а м е ч а н и е.** При сложении большого количества слагаемых, погрешность суммы, вычисленная по формуле (6), оказывается сильно завышенной. В этом случае часто применяют статистическую оценку погрешности суммы:

$\Delta S^* = \sqrt{3n} \cdot 0.5 \cdot 10^{-m}$, где n – число слагаемых в сумме $S^* = a_1^* + a_2^* + \dots + a_n^*$, а m – номер десятичного разряда, до которого округляется результат. Эту формулу используют при $n > 10$.

Относительная погрешность суммы и разности двух чисел будет определяться выражением

$$\delta(a^* \pm b^*) = \frac{\Delta(a^* \pm b^*)}{|a^* \pm b^*|} = \frac{\Delta_{a^*} \pm \Delta_{b^*}}{|a^* \pm b^*|}. \quad (7)$$

Из рассмотренных формул следует, что при сложении или вычитании приближенных чисел желательно, чтобы эти числа обладали одинаковыми абсолютными погрешностями, т.е. одинаковым числом разрядов после десятичной точки. Так, например, при сложении двух приближенных чисел (в записи которых оставлены только верные цифры) $27.412 + 2.3 = 29.712$ в результате получим число с тремя верными цифрами, так как абсолютная погрешность результата ≈ 0.1 .

Следовательно, учет большего числа десятичных разрядов в одном числе при меньшем в другом не приводит к повышению точности результата.

Для абсолютных и относительных погрешностей произведения и частного получим из (5) следующие результаты:

$$\Delta(a^* \cdot b^*) = \Delta_{a^*} |b^*| + \Delta_{b^*} |a^*|, \quad \delta(a^* \cdot b^*) = \delta_{a^*} + \delta_{b^*}, \quad (8)$$

$$\Delta\left(\frac{a^*}{b^*}\right) = \frac{\Delta_{a^*} |b^*| + \Delta_{b^*} |a^*|}{b^{*2}}, \quad \delta\left(\frac{a^*}{b^*}\right) = \delta_{a^*} + \delta_{b^*} \quad (9)$$

Из этих формул следует, что при умножении и делении приближенных чисел, количество значащих цифр целесообразно выравнивать по наименьшему из них.

1.4 Обратная задача теории погрешностей

На практике часто бывает необходимо получить результат вычислений так, чтобы его погрешность не превосходила некоторого допустимого значения. При этом возникает задача определения допустимых (предельных) погрешностей исходных данных задачи, при которых погрешность результата не превысит заданного значения. Эта задача носит название *обратной задачи теории погрешностей*. Для функции одной переменной допустимая погрешность аргумента определяется согласно (4) выражением:

$$\Delta x = \frac{\Delta y^*}{|f'(x^*)|} \quad (10)$$

Для функции нескольких переменных эта задача решается при введении дополнительного предположения – так называемого *принципа равных влияний*.

При этом полагают, что в формуле (5) все слагаемые $\left| \frac{\partial f}{\partial x_i} \right| \cdot \Delta x_i$ равны между собой, тогда

$$\Delta x_i^* = \frac{\Delta y^*}{n \cdot \left| \frac{\partial f}{\partial x_i} \right|}, \quad i = 1, 2, \dots, n. \quad (11)$$

1.5 Особенности машинной арифметики

Причиной появления вычислительных погрешностей является способ представления чисел в компьютере. Современные компьютеры позволяют обрабатывать *целые* и *вещественные* числа. По форме представления, способу хранения и реализации вычислительных операций в процессоре целые и вещественные числа существенно различаются.

Как известно, множество целых чисел бесконечно. Однако процессор в силу ограниченности его *разрядной сетки* может оперировать лишь с некоторым конечным подмножеством этого множества. В современных компьютерах для хранения целого числа обычно отводится 4 байта памяти, что позволяет представить целые числа, находящиеся примерно в диапазоне от $-2 \cdot 10^9$ до $2 \cdot 10^9$.

При решении научных и инженерных задач в основном используются вещественные числа. В памяти компьютера они представлены в форме с *плавающей точкой* (*floating point*). Десятичное число D в этой форме записи имеет вид, $D = \pm m \cdot 10^n$ где m и n – соответственно *мантисса* числа и его *порядок*.

Например, число 357.5 можно записать в виде: $3575 \cdot 10^{-1}$, $0.3575 \cdot 10^3$, $3.575 \cdot 10^2$.

Последняя запись – *нормализованная форма* числа с плавающей точкой.

Обычная же запись числа в виде 357.5 называется формой записи с *фиксированной точкой*.

Такое представление чисел в компьютерах используется только на этапах их ввода или вывода, в то время как хранение и обработка вещественных чисел осуществляется именно в форме с плавающей точкой.

Все сказанное выше в равной степени относится к любой системе счисления с произвольным основанием.

Например, в двоичной системе счисления (в которой, собственно, и проводятся все операции над числами в компьютере) вещественное число можно представить в виде $0.a_1a_2\dots a_k \cdot 10^n$, где a_1, a_2, \dots, a_k – целые числа из диапазона $[0, 1]$.

В силу ограниченности разрядной сетки процессора количество разрядов мантииссы k не может быть сколь угодно большим, величина порядка n также ограничена и находится в пределах от n_1 до n_2 . Это приводит к тому, что множество вещественных чисел, с которым работает компьютер не является бесконечным.

Можно показать, что при заданных k , n_1 и n_2 это множество содержит $N = 2(n_2 - n_1 + 1)2^{k-1} + 1$ чисел.

При этом минимальное и максимальное по модулю из этих чисел имеют значения

$M_0 = 2^{n_1-1}, \dots, M_\infty = (1 - 2^{-k})2^{n_2}$ и называются, соответственно, *машинным нулем* и *машинной бесконечностью*.

Границы порядка n_1 и n_2 определяют ограниченность вещественных чисел по величине, а разрядность k – дискретность распределения их на отрезке числовой оси.

Конечное количество разрядов мантииссы приводит к тому, что при вычислениях с помощью компьютера неизбежны погрешности округления. Так, например, если при 6 десятичных разрядах к числу 0.214724 прибавить число, меньшее по модулю половины последнего разряда (т.е. меньшее по модулю 0.0000005), в результате получится то же самое число 0.214724.

Величина ошибки округления определяется количеством разрядов мантииссы. При округлении по дополнению (которое, как правило, и реализуется в компьютерах) максимальная относительная погрешность округления равна $\varepsilon_{\max} = 0.5 \cdot 2^{1-k}$.

Величина ε_{\max} называется *машинной точностью* или *машинным ε* .

Для чисел, представленных в форматах с одинарной ($k=24$) и двойной ($k=53$)

Пример. Пусть требуется найти сумму пяти четырехразрядных чисел:

$S = 0.2764 + 0.3944 + 1.475 + 26.46 + 1364$. Складывая все эти числа, а затем, округляя результат до четырех значащих цифр, получаем $S = 1393$.

Однако при вычислениях на компьютере округление происходит после каждой операции сложения. Предполагая условно сетку четырех разрядной ($k=4$), проследим за вычислениями на компьютере суммы чисел в порядке их записи:

$0.2764 + 0.3944 = 0.6708$, $0.6708 + 1.475 = 2.156$, $2.156 + 26.46 = 28.62$;
 $28.62 + 1364 = 1393$. Получили $S_1 = 1393$, т.е. верный результат.

Изменим теперь порядок вычислений и начнем складывать числа последовательно от последнего к первому: $1393 + 26.46 = 1390$, $1390 + 1.475 = 1391$,

$1391 + 0.3944 = 1391$, $1391 + 0.2764 = 1391$. Здесь окончательный результат

$S_2 = 1391$, он менее точный.

Анализ процесса вычислений показывает, что потеря точности здесь происходит из-за того, что прибавления к большому числу малых чисел не происходит, поскольку они выходят за рамки разрядной сетки. Таких малых чисел может быть очень много, но на результат они все равно не повлияют, поскольку прибавляются по одному, что и имело место при вычислении S_2 .

Здесь необходимо придерживаться правила, в соответствии с которым сложение чисел следует проводить по мере их возрастания.

Таким образом, в машинной арифметике из-за погрешностей округлений существенен порядок выполнения операций.

Как узнать, насколько адекватные результаты даёт компьютерная модель, которую вы построили, или, как принято говорить у вычислительных математиков, имеется ли сходимость решения модели к решению исходной задачи?

В вычислительной практике большое значение имеет чувствительность решения к малым изменениям входных данных. Задача называется плохо обусловленной, если малые изменения входных данных приводят к заметным изменениям решения. Измерить эту обусловленность на практике очень просто. Нужно просто попробовать чуть-чуть поменять входные данные и посмотреть, как меняется результат. Иначе, нужно выяснить, с какой погрешностью заданы входные данные, и экспериментально проверить в каких пределах меняется результат при варьировании входных данных в пределах их погрешности. Примеры плохо обусловленных задач, выполненных в системе MathCAD представлены ниже.

Пример 1

$$x^4 - 4x^3 + 8x^2 - 16x + 15.9 = C$$

$$\left(\begin{array}{l} 3.129867326134236173210^{-3} - 2.0000073624168891360 \\ 3.129867326134236173210^{-3} + 2.0000073624168891360 \\ 1.8849356568081153331 \\ 2.1088046085396161946 \end{array} \right)$$

$$\left(\begin{array}{l} 2 \cdot i \\ -2 \cdot i \\ 2 \\ 2 \end{array} \right)$$

has solution(s)

$$x^4 - 4x^3 + 8x^2 - 16x + 16 = C$$

Рис. 1

В примере 1 решаются уравнения

$$\begin{aligned}
 x^4 - 4x^3 + 8x^2 - 16x + 15,9 &= 0, \\
 x^4 - 4x^3 + 8x^2 - 16x + 16 &= 0, \\
 x^4 - 4x^3 + 8x^2 - 16x + 15,99999999 &= 0
 \end{aligned}$$

Из рис.1 видно, что решения уравнений, у которых свободные члены 15,9, 16, 15, 999999999 значительно отличаются друг от друга.

Пример 2

Given

$$u + 10v = 11$$

$$100u + 1001v = 1101$$

$$\text{Find}(u, v) \rightarrow \begin{pmatrix} 1 \\ 1 \end{pmatrix}$$

Given

$$u + 10v = 11,1$$

$$100u + 1001v = 110,1$$

$$\text{Find}(u, v) \rightarrow \begin{pmatrix} 101,1 \\ -9,0 \end{pmatrix}$$

Плохо обусловленная задача. Достаточно проверить входные данные

Рис. 2.

В примере 2 приведены решения двух систем

$$\left\{ \begin{array}{l} u + 10v = 11, \\ 100u + 1001v = 1101 \end{array} \right\} , \left\{ \begin{array}{l} u + 10v = 11,1 \\ 100u + 1001v = 110,1 \end{array} \right.$$

А также разные решения этих систем, в которых правые части отличаются на 0,1.

Вопросы для самоконтроля

1. Каковы основные источники погрешностей?
2. Что называется верной цифрой числа?
3. Как по записи приближенного числа оценить его абсолютную и относительную погрешности?
4. Как правильно записывать результат приближенных вычислений или измерений?
5. Что составляет прямую и обратную задачи теории погрешностей?
6. Как определяется погрешность приближенного значения функции многих переменных?
7. Как получить формулы для определения погрешностей арифметических операций?
8. Каковы особенности вычисления сумм приближенных чисел с использованием компьютера?
9. Какие задачи называют плохо обусловленными?
10. Как на практике определить плохую обусловленность задачи?

ГЛАВА 2

РЕШЕНИЕ НЕЛИНЕЙНЫХ УРАВНЕНИЙ

2.1 Понятия и определения

Инженеру, а сейчас и бакалавру при решении практической задачи приходится решать алгебраические и трансцендентные уравнения (нелинейные). Выбор подходящего метода решения зависит от характера рассматриваемой задачи, а его ценность определяется быстротой и эффективностью полученного решения.

Алгебраическими уравнениями называют уравнения, содержащие только алгебраические функции (целые, иррациональные, рациональные).

Определение.

Уравнение вида

$$a_0x^n + a_1x^{n-1} + a_2x^{n-2} + \dots + a_{n-1}x + a_nx^0 = 0, \quad (2.1)$$

где $a_0, a_1, a_2, \dots, a_{n-1}, a_n$ некоторые числа называется алгебраическим. Число n называют степенью алгебраического уравнения.

Трансцендентными уравнениями называют уравнения, содержащие тригонометрические, логарифмические, показательные и другие функции.

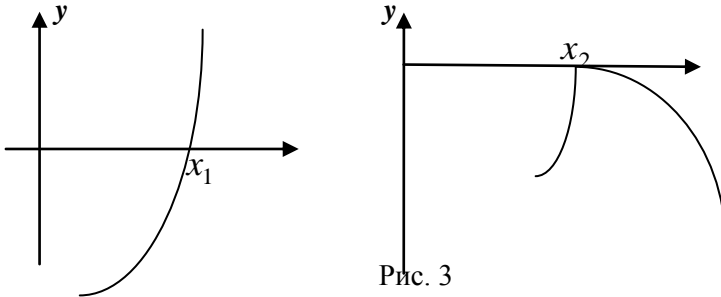
Например, уравнение $2x^5 - x^3 + 4x = x - 6$, является алгебраическим, а уравнения $12\log_{x-1}(2-x)^5 = 10$, $\operatorname{tg}(x-2) - 15x = 0$ - трансцендентными.

В общем виде запишем нелинейное уравнение в виде

$$f(x) = 0. \quad (2.2)$$

Определение. Корнем (решением) уравнения (2) называется такое значение x , которое, будучи подставленным в исходное уравнение обращает его в тождество.

Графически корень уравнения (2.2) представляет собой абсциссу точки пересечения, касания или другой общей точки графика функции $y = f(x)$ и оси ox (рис.3).



Методы решения нелинейных уравнений делятся на две группы:

- точные методы;
- приближенные методы.

Точные методы позволяют записать корни в виде некоторого конечного соотношения (формулы), например, формула нахождения корней квадратного уравнения. Из курса средней школы

известны формулы для нахождения корней тригонометрических, логарифмических, показательных и простейших алгебраических уравнений.

Как известно, многие уравнения и системы уравнений не имеют аналитических решений. Но, построив график такой функции, можно увидеть ее точки пересечения с осью ox , значит, корни уравнения существуют.

Математики Абель (1802-1829) и Галуа (1811-1832) доказали, что нельзя построить формулу, по которой можно было бы решить произвольное алгебраическое уравнение степени, выше четвертой.

Кроме этого, в некоторых уравнениях содержатся коэффициенты, значения которых известны приблизительно, следовательно, нельзя найти аналитическое решения такого уравнения. Для решения подобных уравнений существуют итерационные (приближенные) методы решения с заданной степенью точности.

При численном подходе задача о решении нелинейных уравнений разбивается на два этапа: *локализация* (отделение) корней, т.е. нахождение таких отрезков на оси x , в пределах которых содержится один единственный корень, и *уточнение корней*, т.е. вычисление приближенных значений корней с заданной точностью.

2.2. Локализация (отделение) корней уравнения

Пусть дано уравнение $f(x) = 0$. По теореме Больцана-Коши можем утверждать, что, если функция $y = f(x)$ непрерывна и монотонна на отрезке $[a, b]$,

а на концах отрезка ее значения имеют разные знаки

$$f(a) \cdot f(b) < 0, \quad (2.3)$$

то на рассматриваемом отрезке имеется корень уравнения и этот корень единственный.

В качестве признака монотонности функции можно воспользоваться условием знакопостоянства первой производной $y'(x)$ на этом отрезке или *достаточным условием* монотонности функции на отрезке:

Если первая производная функции на данном отрезке $f'(x) > 0$, то функция возрастает на этом отрезке, если $f'(x) < 0$, то функция убывает. Это условие можно использовать для аналитического отделения корней уравнения, находя интервалы монотонности функции. Отделение корней можно выполнить *графически*, если построить график функции $y = f(x)$. Графический метод отделения корней больше распространен в инженерной практике. В ряде случаев бывает удобно заменить это уравнение эквивалентным уравнением вида $f_1(x) = f_2(x)$. Корни этого уравнения определяются абсциссами точек пересечения графиков функций $y = f_1(x)$ и $y = f_2(x)$.

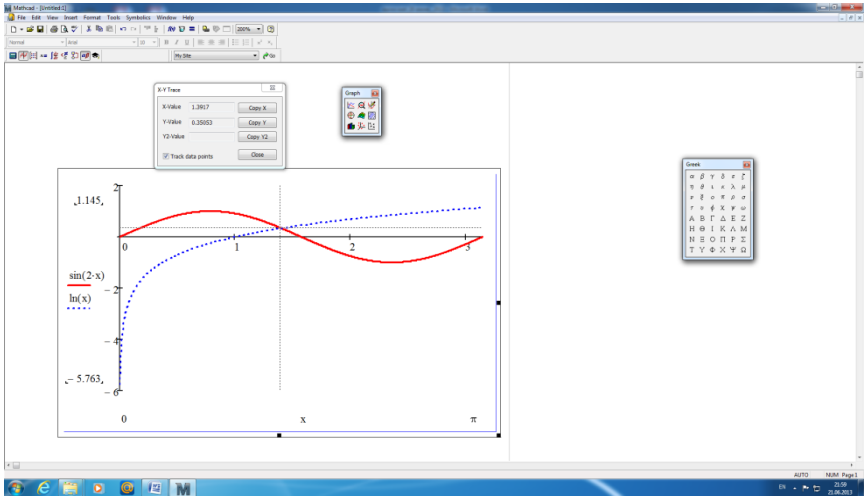


Рис. 4

В качестве примера рассмотрено уравнение $\sin 2x - \ln x = 0$. Переходя к эквивалентному уравнению $\sin 2x = \ln x$ построены графики функций $y = \sin 2x$ и $y = \ln x$ (рис. 4).

Графики построены при помощи системы MathCAD. В окне «X-Y Trace» изображаются координаты точек пересечения графиков $y = \sin 2x$ и $y = \ln x$:

X-Value 1.405

Y-Value 0.32555.

Корень уравнения $\sin 2x - \ln x = 0$ приблизительно равен $x=1.405$, т.е. он находится в интервале $[1,2]$. (Правую границу тоже можно уточнить при помощи MathCAD).

2.3 Уточнение корней уравнения

Уточнение приближенных корней - вычисление корня из выбранного промежутка с заданной степенью точности, т.е. с заданной погрешностью ε . Это означает, что вычисленное значение корня x должно отличаться от точного x^* не более чем на величину ε :

$$|x^* - x| \leq \varepsilon.$$

Процедура численного определения приближенных значений корней нелинейных уравнений, как правило, состоит в выборе начального приближения к корню $x_0 \in [a, b]$, которое можно определить локализацией и вычислении по некоторой формуле последующих приближений x_1, x_2, \dots и т.д. Каждый такой шаг называется итерацией (от латинского *iteratio* – повторение), а сами методы уточнения – *итерационными* методами.

В результате *итераций* получается последовательность приближенных значений корня $x_0, x_1, x_2, \dots, x_k$, которая называется итерационной последовательностью. Если эти значения с ростом k стремятся к точному значению

корня x^* , т.е. существует

$$\lim_{k \rightarrow \infty} \{x_k\} = x^* \quad (2.4),$$

то говорят, что итерационный процесс *сходится*.

Сходимость итерационного процесса означает, что погрешность каждого последующего приближения должна быть меньше погрешности предыдущего приближения, т.е. погрешность приближенных значений с каждым шагом должна уменьшаться:

$$|x^* - x_{k+1}| < |x^* - x_k|.$$

В общем случае это неравенство можно представить в виде:

$$|x^* - x_{k+1}| < q|x^* - x_k|^\alpha. \quad (2.5),$$

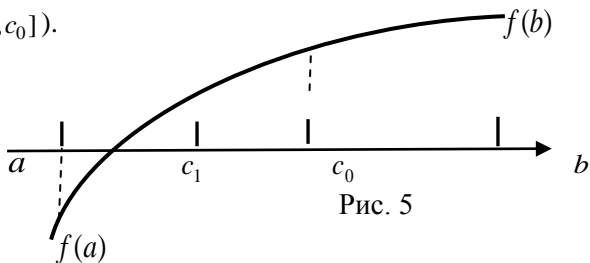
где $q > 0$ и $\alpha \geq 1$ – некоторые числа, значения которых определяются методом уточнения корня. От значений q и α зависит насколько с каждым шагом уменьшается погрешность приближенных значений и, соответственно, насколько быстро можно получить приближенное значение с заданной точностью.

Главным показателем скорости сходимости метода является значение α , называемое *порядком сходимости*. При $\alpha = 1$ погрешность с каждым шагом убывает линейно, в этом случае говорят о *линейной сходимости*. Если $\alpha > 1$ то говорят, что имеет место *сверхлинейная сходимость*. Существуют разные методы уточнения корней.

2.4 Методы уточнения корней

2.4.1. Метод половинного деления (бисекции, дихотомии)

Считаем, что отделение корней уравнения $f(x) = 0$ проведено и на отрезке $[a, b]$ расположен один корень, который необходимо уточнить с погрешностью ε . В качестве начального приближения корня принимаем середину этого отрезка: $c_0 = \frac{a+b}{2}$ (рис. 5). Затем исследуем значения функции на концах отрезков $[a, c_0], [c_0, b]$. Тот из отрезков, на концах которого $f(x)$ принимает значения разных знаков, содержит искомый корень; поэтому его принимаем в качестве нового отрезка $[a_1, b_1]$ (на рис. 5 это отрезок $[a, c_0]$).



Вторую половину отрезка $[a, b]$, на которой $f(x)$ не меняет знак, отбрасываем. В качестве следующего приближения корня принимаем середину нового отрезка $c_1 = \frac{a_1 + b_1}{2}$ и т.д. Таким образом, k -е приближение вычисляется как $c_k = \frac{a_k + b_k}{2}$. После каждой итерации отрезок, на котором расположен корень, уменьшается вдвое, а после k итераций в 2^k раз: $b_k - a_k = \frac{b - a}{2^k}$.

Прекратить итерационный процесс следует, когда будет достигнута заданная точность, т.е. при выполнении условия $|x^* - c_k| < \varepsilon$. Поскольку корень x^* принадлежит отрезку $[a_k, b_k]$, а c_k — середина этого отрезка, то величина $|x^* - c_k|$ всегда будет меньше половины длины отрезка $[a_k, b_k]$ (см. рис. 5), т.е. $|x^* - c_k| < \frac{|b_k - a_k|}{2}$. Следовательно, условие $|x^* - c_k| < \varepsilon$ будет выполнено, если

$$|b_k - a_k| < 2\varepsilon \quad (2.6)$$

Таким образом, итерационный процесс нужно продолжать до тех пор, пока не будет выполнено условие (2.6).

В отличие от большинства других методов уточнения, метод половинного деления сходится всегда, т.е. обладает *безусловной* сходимостью. Кроме этого он чрезвычайно прост, поскольку требует лишь вычисления значений функции $f(x)$ и, поэтому, применим для решения любых уравнений.

Однако метод половинного деления довольно медленный. С каждым шагом погрешность приближенного значения уменьшается в два раза, т.е. $|x^* - c_{k+1}| \leq \frac{1}{2}|x^* - c_k|$, поэтому данный метод является методом с линейной сходимостью.

Вычислим количество итераций N , требуемое для достижения заданной точности ε . Пользуясь выражением $b_k - a_k = \frac{b - a}{2^k}$.

можно выяснить для каких значений k будет выполнено условие (6), и взять в качестве N наименьшее из таких k :

$k > \log_2 \frac{b-a}{2\varepsilon} \Rightarrow N = \text{int}(\log_2 \frac{b-a}{2\varepsilon}) + 1$, где $\text{int}(x)$ - целая часть числа x .

Например, при $b-a=1$ и $\varepsilon=10^{-6}$ получаем $\Rightarrow N = \text{int}(\log_2 \frac{1}{2 \cdot 10^{-6}}) + 1 = \text{int}(18,932+1) = 19$, $\Rightarrow N = 19$.

➤ **З а м е ч а н и е**

При реализации метода следует учитывать, что функция $f(x)$ вычисляется с некоторой абсолютной погрешностью Δf . Вблизи корня значения функции $f(x)$ малы по абсолютной величине и могут оказаться сравнимы с погрешностью ее вычисления. Другими словами, при подходе к корню мы можем попасть в полосу шумов $2\Delta f$ и дальнейшее уточнение корня окажется невозможным. Поэтому целесообразно задать ширину полосы шумов и прекратить итерационный процесс при попадании в нее. Если принять $\Delta f = \varepsilon$, то итерационный процесс можно завершать, когда значение функции $f(x)$ после k -й итерации станет меньшим по модулю ε , т.е. $|f(c_k)| < \varepsilon$. Также необходимо иметь в виду, что при уменьшении интервала $[a, b]$ увеличиваются погрешности вычисления его длины $b-a$ за счет вычитания близких чисел.

Пример 1.

Найти корни алгебраического уравнения $x^3 - 2x - 5 = 0$ методом половинного деления средствами MathCAD с точностью $2 \cdot \varepsilon = 0.02$

Решение.

Локализуем корни уравнения графически, для чего построим график функции $x^3 - 2x - 5 = 0$ так, чтобы «увидеть» точки пересечения графика с осью ox , изменяя, по необходимости масштаб графика. Из рисунка ниже видно, на каком участке $[1.5; 3]$ следу-

ет искать единственный корень. Вычисляем значения функции на концах отрезка, проверяя условие $f(a) \cdot f(b) < 0$

$$f(a) = f(1.5) = -4.625, \text{ а } f(b) = f(3) = 16.$$

Значения функции на концах отрезка имеют разные знаки, поэтому находим первое приближение функции по формуле

$$c_1 = \frac{a+b}{2} = \frac{1.5+3}{2} = 2.25, \quad c_1 = 2.25.$$

Находим значение функции в этой точке $f(2.25) = 1.829 \neq 0 \Rightarrow$ уточняем корень далее. Кроме того, учитывая, что $f(2.25) > 0$, отрезок сужаем, оставляя левый конец прежним, а правый заменим на 2.25. Таким образом, имеем $[a_1; b_1] = [1.5; 2.25]$.

$$\text{Находим } c_2 = \frac{a_1 + b_1}{2} = \frac{1.5 + 2.25}{2} = 1.875, \quad c_2 = 1.875.$$

$f(1.875) = -2.158 \neq 0$, значит, левый конец отрезка вместо 1.5 нужно взять значение 1.875.

Итак, $[a_2; b_2] = [1.875; 2.25]$. Уточняем корень дальше.

Еще раз делим отрезок пополам

$$c_3 = \frac{a_2 + b_2}{2} = \frac{1.875 + 2.25}{2} = 2.063, \quad c_3 = 2.063.$$

Формат результата в MathCAD выводит значение с точностью до трех знаков после запятой). $f(2.063) = -0.375 \neq 0$.

Значение функции в этой точке близко к нулю. Погрешность вычислений

$$|b_k - a_k| < 2\varepsilon \text{ равна } |b_2 - a_2| = |2.25 - 1.875| = 0.375 > 2 \cdot \varepsilon, \quad 2 \cdot \varepsilon = 0.02$$

Находя дальнейшие приближения корня, найдем, что седьмое приближение равно 2.099 и значение функции в этой точке близко к нулю $f(2.099) = 0.044$.

Значит, нашли корень уравнения с заданной точностью.

Погрешность вычислений равна

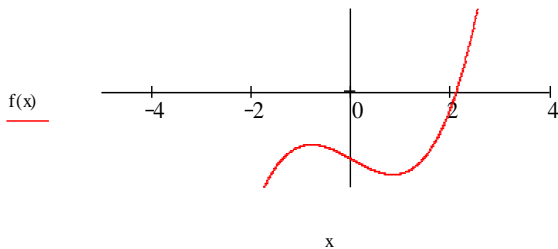
$$|b_k - a_k| < 2\varepsilon, \quad |b_7 - a_7| = 0.012 < 0.02 = 2\varepsilon$$

Постановка задачи. Решить уравнение $x^3 - 2x - 5 = 0$ методом половинного деления с точностью $\varepsilon = 0.01$.

Решение

1. Локализуем корни уравнения графическим методом

$$f(x) := x^3 - 2x - 5$$



из графика следует, что один корень есть на отрезке $[a, b] = [1.5; 2.5]$

$$a := 1.5 \quad b := 2.5 \quad \varepsilon := 0.01$$

заданное значение точности приближения

$$f(x) := x^3 - 2x - 5 \quad f(a) = -4.625 \quad f(b) = 1.875$$

1. найдем первое приближение корня c_1

$$c_1 := \frac{a + b}{2} \quad c_1 = 2.25 \quad f(c_1) = 1.891$$

т.к. $f(c_1) > 0$, то новый отрезок $[a_1, b_1] = [1.5; 2.25]$.

$$a_1 := 1.5 \quad b_1 := 2.25$$

Проверим условие

$$|b_1 - a_1| < 2 \cdot \varepsilon \quad |b_1 - a_1| = 0.75$$

условие нарушено

$$2 \cdot \varepsilon = 0.02$$

Рис. 6

- 2 найдем второе приближение
корня c_2

$$c_2 := \frac{a_1 + b_1}{2} \quad c_2 = 1.875 \quad f(c_2) = -2.158$$

т.к. $f(c_2) < 0$, то новый отрезок $[a_2, b_2] = [1.875; 2.25]$

$$a_2 := 1.875 \quad b_2 := 2.25$$

Проверим условие $|b_2 - a_2| < 2 \cdot \varepsilon$ $|b_2 - a_2| = 0.375$

условие нарушено $2 \cdot \varepsilon = 0.02$

3. найдем третье приближение
корня c_3

$$c_3 := \frac{a_2 + b_2}{2} \quad c_3 = 2.063 \quad f(c_3) = -0.351$$

т.к. $f(c_3) < 0$, то новый отрезок $[a_3, b_3] = [2.063; 2.25]$ $b_3 := 2.25$

Проверим условие

$$|b_3 - a_3| < 2 \cdot \varepsilon \quad |b_3 - a_3| = 0.187$$

$$2 \cdot \varepsilon = 0.02$$

условие нарушено

4. найдем четвертое
приближение корня c_4

$$c_4 := \frac{a_3 + b_3}{2} \quad c_4 = 2.157 \quad f(c_4) = 0.716$$

т.к. $f(c_4) > 0$, то новый отрезок $[a_4, b_4] = [2.063; 2.157]$

$$a_4 := 2.063 \quad b_4 := 2.157$$

Проверим условие

$$|b_4 - a_4| < 2 \cdot \varepsilon \quad |b_4 - a_4| = 0.094$$

условие нарушено $2 \cdot \varepsilon = 0.02$

Рис. 7

5. найдем пятое приближение корня c_5 $c_5 := \frac{a_4 + b_4}{2}$

$$c_5 = 2.11 \quad f(c_5) = 0.174$$

т.к. $f(c_5) > 0$, то новый отрезок $[a_5, b_5] = [2.063; 2.11]$ $a_5 := 2.063$ $b_5 := 2.11$

Проверим условие $|b_5 - a_5| < 2 \cdot \varepsilon$ $|b_5 - a_5| = 0.047$

6. найдем шестое приближение корня c_6 $c_6 := \frac{a_5 + b_5}{2}$ $c_6 = 2.087$ $f(c_6) = -0.089$ условие нарушено

т.к. $f(c_6) < 0$, то новый отрезок $[a_6, b_6] = [2.087; 2.11]$ $a_6 := 2.087$ $b_6 := 2.11$

Проверим условие $|b_6 - a_6| < 2 \cdot \varepsilon$ $|b_6 - a_6| = 0.023$

погрешность вычислений превышает заданную точность 0.02

7. найдем седьмое приближение корня c_7 $c_7 := \frac{a_6 + b_6}{2}$ $c_7 = 2.099$ $f(c_7) = 0.044$

т.к. $f(c_7) > 0$, то новый отрезок $[a_7, b_7] = [2.087; 2.099]$

$$a_7 := 2.087 \quad b_7 := 2.099$$

Проверим условие $|b_7 - a_7| < 2 \cdot \varepsilon$

$$|b_7 - a_7| = 0.012$$

Условие выполнено: $0.012 < 0.02$ $2 \cdot \varepsilon = 0.02$

Ответ: $x = 2.099$

Рис. 8

На рис. 6-8 представлены все шаги алгоритма решения данной задачи в универсальной математической системе Mathcad.

2.4.2 Метод хорд

Рассматриваемый метод так же, как и метод половинного деления, предназначен для уточнения корня на интервале $[a, b]$, на концах которого функция $f(x)$ принимает значения разных знаков. Очередное приближение в отличие от метода половинного деления берем не в середине отрезка, а в точке x_0 , где пересекает ось абсцисс прямая линия (хорда), проведенная через точки A и B (рис. 4). Запишем уравнение прямой, проходящей через точки A и B

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

Для точки пересечения прямой с осью абсцисс $x = x_0, y = 0$ получим уравнение

$$(2.7) \quad x_0 = a - \frac{b - a}{f(b) - f(a)} f(a)$$

В качестве нового интервала для продолжения итерационного процесса выбираем тот из двух $[a, x_0]$ и $[x_0, b]$, на концах которого функция $f(x)$ принимает значения разных знаков. Для рассматриваемого случая (рис. 4) выбираем отрезок $[a, x_0]$, так как $f(a)f(b) > 0$.

Следующая итерация состоит в определении нового приближения x_1 как точки пересечения хорды AB_1 с осью абсцисс и т.д. Заканчиваем процесс

уточнения корня, когда расстояние между очередными приближениями станет меньше заданной точности, т.е.

$$|x_{k+1} - x_k| < \varepsilon \quad (2.8)$$

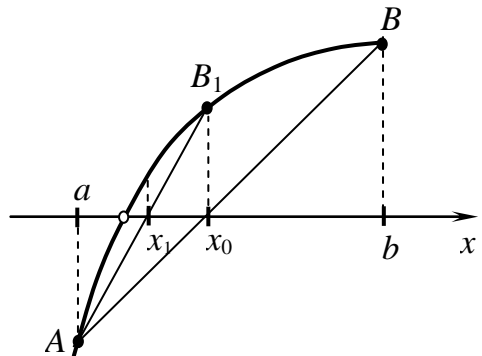


Рис. 9. Метод хорд.

или при выполнении условия (2.8).

➤ З а м е ч а н и е

Метод половинного деления и метод хорд очень похожи, в частности, процедурой проверки знаков функции на концах отрезка. При этом второй их них в ряде случаев дает более быструю сходимость итерационного процесса. Однако в некоторых случаях метод хорд может сходиться существенно медленнее метода половинного деления. Оба рассмотренных метода не требуют знания дополнительной информации о функции $f(x)$. Например, не требуется, чтобы функция была дифференцируема.

Даже для разрывных функций рассмотренные методы обладают рассмотренные методы обладают гарантированной сходимостью.

Представим алгоритм вычислений с элементами программирования. В системе MathCAD-программы с точки зрения программиста представляют собой подпрограммы-функции, которые могут возвращать в качестве результата число, вектор или матрицу. Функции могут вызывать сами себя (рекурсивно определенные функции) или другие подпрограммы-функции, определенные выше в том же MathCAD-документе.

Эти подпрограммы-функции составляются так же, как и определения функций. Рассмотрим простой пример вычисления суммы: $Sum(n)=1+2+3+..+n$:

Задайте сначала имя функции Sum, вслед за которым введите список параметров (n) и оператор присваивания :=

Введите начальное значение $s \leftarrow 0$ и отметьте его целиком. Стрелка служит оператором присваивания в теле программы, вводятся клавишей { (открывающая фигурная скобка) или выбором кнопки со стрелкой с панели.

Создайте новую строку программы (вставка вертикальной черты) при помощи кнопки Add Line (клавиша -) (закрывающая квадратная скобка)). На экране появится вертикальная черта и пустая ячейка.

Выполните щелчок на кнопке for, чтобы ввести оператор цикла. Перед символом принадлежности введите переменную i, а

после него - промежуток изменения переменной 1..n. В находящейся ниже ячейке введите присваивание $s \leftarrow s+i$.

Ниже в таблице 1 приведены операторы, находящиеся на кнопках панели программирования.

Таблица 1

Команды панели Symbolic (символы)		
Команды	Функция	Пример
Add Line	Добавляет новую строку под/над (зависит от выделения) текущей строкой.	
\leftarrow	Присваивание значения локальной переменной.	$y \leftarrow 0$
If	Условный оператор (оператор ветвления)if; условие должно стоять после if, а оператор, который исполняется, если выполнено заданное условие, - перед if.	$f(x) := \begin{cases} -x & \text{if } x < 0 \\ x & \text{otherwise} \end{cases}$ $f(-3) = 3 \quad f(4) = 4$
Otherwise	Обозначает оператор, который должен быть исполнен, если условие оператора if не выполняется.	$f(x) := \begin{cases} -x & \text{if } x < 0 \\ x & \text{otherwise} \end{cases}$ $f(-3) = 3 \quad f(4) = 4$
For	Цикл for; за ключевым словом for следует переменная-счетчик, а после символа принадлежности вводится промежуток изменения этой переменной. Внутренние операторы цикла сдвинуты немного	$Sum(n) := \begin{cases} s \leftarrow 0 \\ \text{for } i \in 1..n \\ s \leftarrow s + i \end{cases}$

вправо.

$$Sum(4) = 10$$

While	Цикл <code>while</code> ; внутренние операторы цикла будут исполняться до тех пор, пока будет истинным условие, следующее за ключевым словом <code>while</code> . Пример показывает применение цикла для нахождения нулей функции методом касательных Ньютона.	$N(x, f, f_x) :=$ $while f(x) > 10^{-6}$ $x \leftarrow x - \frac{f(x)}{f_x(x)}$ $N(2, \sin, \cos) = 3.142$
Break	Служит для преждевременного завершения цикла, чтобы, например, избежать закливания или слишком продолжительных вычислений.	$break \text{ if } i \geq 10$
Continue	Служит для преждевременного завершения текущей итерации цикла; сам цикл при этом продолжается.	$continue \text{ if } x \geq 1$
Return	Преждевременное завершение программы; указанное в ячейке значение будет возвращено.	$return \ y$
On error	Если при вычислении выражения <code>expr2</code> возникла ошибка, вычисляется выражение <code>expr1</code> .	$expr1 \text{ on error}$ $expr2$

Функция *bs_root* (*bisection root*) содержит четыре формальных параметра: f – функция, нули которой ищутся; a и b – границы отрезка локализации корня; ε – абсолютная погрешность. Основной алгоритм реализован с помощью циклического оператора *while*, в теле которого происходит вычисление средней точки от-

резка $[a, b]$, проверка знаков функции на отрезке $[a, c]$ и выбор нового отрезка. Для подсчета числа итераций используется переменная it . В качестве результата функция возвращает приближенное значение корня и количество итераций. Поскольку в программах-функциях пакета Mathcad в качестве результата можно указывать только одну переменную, то для получения нескольких значений, в качестве итоговой переменной может служить вектор или матрица.

В приведенном ниже примере формируется вектор-строка ans , первый (нулевой) элемент которой содержит приближенное значение, а второй количество итераций (рис.9)

$$\begin{array}{l}
 bs_root(f, a, b, \varepsilon) := \left| \begin{array}{l}
 it \leftarrow 0 \\
 \text{while } |b - a| > 2 \cdot \varepsilon \\
 \left| \begin{array}{l}
 it \leftarrow it + 1 \\
 c \leftarrow \frac{(a + b)}{2} \\
 b \leftarrow c \text{ if } f(a) \cdot f(c) < 0 \\
 a \leftarrow c \text{ otherwise}
 \end{array} \right. \\
 ans_{0,0} \leftarrow c \\
 ans_{0,1} \leftarrow it \\
 ans
 \end{array} \right.
 \end{array}$$

Рис. 9

Программа `roots` реализует алгоритм табличного поиска корней и содержит следующие параметры: f – функция; a и b – границы интервала поиска; h – шаг поиска; ε – точность.

В теле программы используются следующие переменные: n_root – номер найденного корня (нумеруются с нуля); $x1$ и $x2$ – границы текущего отрезка локализации; $y1$ и $y2$ – значения функции $f(x)$ на границах отрезка локализации; bs – приближенное значение корня, полученное методом половинного деления (рис. 10)

```

roots(f , a , b , h , ε) :=
  if a > b ∨ h ≤ 0 ∨ ε ≤ 0
    | error("Некорректные параметры")
    | break
  n_root ← 0
  x1 ← a
  y1 ← f(x1)
  while 1
    | x2 ← x1 + h
    | y2 ← f(x2)
    | if (y1 · y2) < 0
      | | bs ← bs_root(f , x1 , x2 , ε)
      | | ansn_root, 0 ← bs0, 0
      | | ansn_root, 1 ← bs0, 1
      | | n_root ← n_root + 1
    | break if x2 > b
    | x1 ← x2
    | y1 ← y2
  ans

```

Рис. 10

Поясним основные шаги алгоритма.

На начальном этапе проверяется корректность исходных данных. В случае ошибки работа программы прекращается с выдачей соответствующего сообщения.

Поиск корней начинается с точки $x1=a$. В этой точке вычисляется значение функции $y1$. В теле цикла *while* вычисляется правая граница отрезка локализации $x2$ и значение функции в этой точке $y2$. Далее проверяются знаки функции на концах отрезка $[x1,x2]$. Если знаки противоположные (произведение $y1$ и $y2$ меньше нуля), то вызывается функция уточнения корня *bs_root* и формируется вектор результатов *ans*. Номер корня увеличивается на единицу.

Для перехода к следующему отрезку поиска осуществляется переписывание $x1 \leftarrow x2, y1 \leftarrow y2$.

Цикл прекращается если $x2$ превысит значение правой границы интервала поиска b .

Далее приведен пример решения уравнения непосредственно при помощи вычислительной формулы.

Пример 2.

Найти корни алгебраического уравнения

$x^3 - 0.2 \cdot x^2 - 0.3 \cdot x + 1.3 = 0$ методом хорд с заданной точностью $\varepsilon = 0.01$ средствами MathCAD (без средств программирования).

Решение.

Отделим корни уравнения графически, как в рассмотренном примере 1. Далее на найденном отрезке $[-2; 0]$ уточняем корни по формуле (2.7): $x_0 = a - \frac{b-a}{f(b)-f(a)} f(a)$ с требуемой точностью

$\varepsilon = 0.01$, найденной по формуле $|x_{k+1} - x_k| < \varepsilon$.

Подробное описание алгоритма представлено ниже в документах MathCAD. (Рис. 11-13).

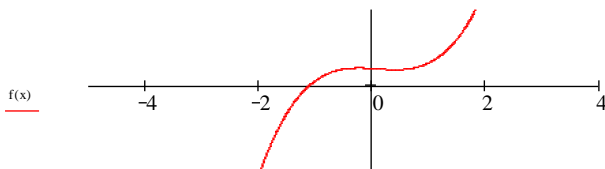
Кроме этого, возможно написание процедур, позволяющих решить уравнение методом хорд.

Постановка задачи. Решить уравнение $x^3 - 0.2x^2 - 0.3x + 1.3 = 0$ методом хорд с точностью $\epsilon = 0.01$.

Решение

Локализуем корни уравнения графическим методом

$$f(x) := x^3 - 0.2x^2 - 0.3x + 1.3$$



из графика следует, что один корень есть на отрезке $[a, b] = [-2; 0]$

$$a := -2 \quad b := 0 \quad \epsilon := 0.01 \quad \text{задание точности и приближения}$$

$$f(a) = -6.9 \quad f(b) = 1.3$$

1. найдем первое приближение корня

$$x_0 := a - \frac{b - a}{f(b) - f(a)} \cdot f(a)$$

$$f(x_0) = 1.343$$

$$x_0 = -0.317$$

т.к. $f(x_0) > 0$, то новый отрезок $[a, b_1] = [-2; -0.317]$.

$$b_1 := -0.317$$

$$b := b_1$$

2 найдем второе приближение корня

$$x_1 := a - \frac{b - a}{f(b) - f(a)} \cdot f(a)$$

проверим точность вычисленного корня

$$x_1 = -0.591$$

$$f(x_1) = 1.201$$

$$|x_1 - x_0| = 0.274$$

это значение больше заданной точности 0.01

Рис. 11. Начало документа Mathcad

3. Выберем новый отрезок. Т.к. $f(x_1) > 0$, то новый отрезок $[a, b_2] = [-2; -0.591]$. Найдем третье приближение

$$b := b_2$$

проверим точность вычисленного $x_2 := a - \frac{b-a}{f(b)-f(a)} \cdot f(a)$ $x_2 = -0.8$
 корня $|x_2 - x_1| = 0.209$

это значение $f(x_2) = 0.9$
 больше заданной
 точности 0.01

4. Выберем новый отрезок. Т.к. $f(x_2) > 0$, то новый отрезок $[a, b_3] = [-2; -0.8]$. Найдем четвертое приближение корня

$$b_3 := -0.8 \quad b := b_3$$

$$x_3 := a - \frac{b-a}{f(b)-f(a)} \cdot f(a) \quad x_3 = -0.938$$

проверим точность вычисленного
 корня

$$|x_3 - x_2| = 0.139$$

видим, что точность ближе к заданному $f(x_3) = 0.579$
 значению 0.01, но еще не меньше его

5. Выберем новый отрезок. Т.к. $f(x_3) > 0$, то новый отрезок $[a, b_4] = [-2; -0.938]$. Найдем пятое приближение корня

$$b_4 := -0.938$$

$$b := b_4$$

$$x_4 := a - \frac{b-a}{f(b)-f(a)} \cdot f(a)$$

проверим точность вычисленного
 корня

$$|x_4 - x_3| = 0.082$$

$$x_4 = -1.02$$

$$f(x_4) = 0.336$$

видим, что точность вычислений еще не достигнута

6. Выберем новый отрезок. Т.к. $f(x_4) > 0$, то новый отрезок $[a, b_5] = [-2; -1.02]$. Найдем шестое приближение корня $b_5 := -1.02$

$$x_5 := a - \frac{b-a}{f(b)-f(a)} \cdot f(a) \quad b := b_5$$

$$|x_5 - x_4| = 0.045$$

$$x_5 = -1.066$$

точность вычислений не достигнута $f(x_5) = 0.183$

Рис. 12. Продолжение документа Mathcad

7. Выберем новый отрезок. Т.к. $f(x_5) > 0$, то новый отрезок $[a, b_5] = [-2; -1.066]$. Найдем седьмое приближение корня

$$b_6 := -1.066 \quad b := b_6$$

$$x_6 := a - \frac{b - a}{f(b) - f(a)} \cdot f(a) \quad x_6 = -1.09$$

Аналогично вычисляем до тех пор, пока не добьемся требуемой точности

$$|x_6 - x_5| = 0.024 \quad f(x_6) =$$

$$b_7 := -1.09 \quad b := b_7$$

$$x_7 := a - \frac{b - a}{f(b) - f(a)} \cdot f(a) \quad f(x_7) = 0.048$$

$$|x_7 - x_6| = 0.012$$

$$x_8 := a - \frac{b - a}{f(b) - f(a)} \cdot f(a) \quad b_8 := -1.102 \quad b := b_7$$

$$|x_8 - x_7| = 0 \quad x_8 = -1.102$$

Ответ. $x = -1.102$ с точностью до 0.01

Рис. 13. Окончание документа Mathcad

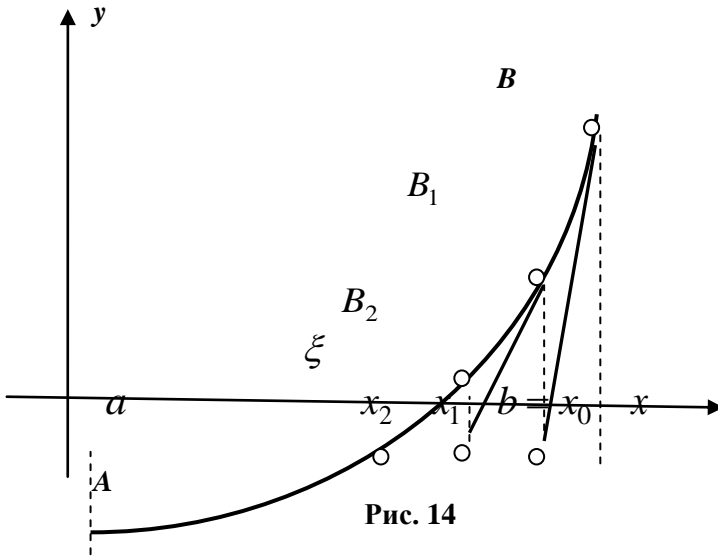
2.4.3. Метод Ньютона (метод касательных)

Более сложные методы уточнения корня используют дополнительную информацию о функции $y = f(x)$, прежде всего, *свойство дифференцируемости*. Как результат они обычно обладают более быстрой сходимостью, но в то же время, применимы для более узкого класса функций, и их сходимость не всегда гарантирована. Примером такого метода служит метод Ньютона.

В данном методе на каждом шаге итерации проводится касательная к кривой $y = f(x)$ при $x = x_i$ и ищется точка пересечения касательной с осью абсцисс (рис.4).

Уравнение касательной можем записать, как уравнение прямой, проходящей через данную точку (x_0, y_0) , с данным угловым коэффициентом k :

$$y - y_0 = k(x - x_0).$$



На рис. 14 изображена рассматриваемая кривая, в точке B с координатами $(x_0, f(x_0))$ проведена касательная до пересечения с осью ox . При этом конец промежутка $[a, b]$ точку $b = x_0$ будем считать начальным приближением корня.

Запишем $y - f(x_0) = y'(x_0)(x - x_0)$, зная, что угловый коэффициент касательной k равен производной функции $y'(x_0)$, вычис-

ленной в точке, куда проведена касательная: $k = y'(b)$. Полагая

$$y = 0, \text{ находим } x = x_0 - \frac{f(x_0)}{f'(x_0)}.$$

Таким образом, точка x_1 пересечения касательной с осью OX равна

$$x_1 = x_0 - \frac{f(x_0)}{f'(x_0)}.$$

Каждые последующие приближения могут быть найдены аналогично, как точки пересечения касательных, проведенных в точках B_1, B_2 и т.д. В методе касательных $i+1$ -е приближение имеет вид:

$$x_{i+1} = x_i - \frac{f(x_i)}{f'(x_i)}. \quad (2.10)$$

Применяя метод Ньютона, следует руководствоваться *следующим правилом*:

За начальное приближение принимается такое значение x_0 из отрезка $[a, b]$, для которого выполняется условие Фурье:

$$f(x_0) \cdot f''(x_0) > 0. \quad (2.11)$$

Сходимость метода Ньютона

Если функция $y = f(x)$ имеет отличную от нуля производную $f'(x)$ на отрезке $[a, b]$, то оценка абсолютной погрешности при

вычислениях вычисляется по формуле $|\varepsilon - x_{i+1}| \leq \frac{f'(x_{i+1})}{\mu}$, где

$$\mu = \min_{a \leq x \leq b} |f'(x)|.$$

На практике применяют чаще формулу $|x_i - x_{i-1}| < \varepsilon$, где ε - заданная точность.

Теперь обратимся к реализации алгоритма метода Ньютона при помощи программирования в MathCAD (рис. 15).

```

n_root(f, a, b, ε) :=
    it ← 0
    max_it ← 1000
    x0 ← a if f(a) ·  $\frac{d^2}{da^2} f(a) > 0$ 
    x0 ← b otherwise
    while 1
        it ← it + 1
        x1 ← x0 -  $\frac{f(x0)}{\frac{d}{dx0} f(x0)}$ 
        break if (|x1 - x0| < ε) ∨ (it > max_it)
        x0 ← x1
    error("Нет сходимости!") if it > max_it
    ans0,0 ← x1
    ans0,1 ← it
    ans

```

Рис. 15

Поскольку метод Ньютона обладает условной сходимостью, т.е. при неблагоприятных обстоятельствах итерации могут и не привести к уточнению корня, то во избежание заикливания вычислительный алгоритм должен учитывать такую возможность. В частности, можно определить максимально допустимое количество итераций, превышение которого будет расценено как отсут-

ствие сходимости и приведет к прекращению вычислений с выводом соответствующего сообщения.

Функция *n_root* имеет те же, формальные параметры, что и функция *bs_root*. Поясним основные шаги алгоритма.

После определения переменных *it* и *it_max*, с помощью операторов *if* и *otherwise* осуществляется выбор начального приближения к корню x_0 .

В качестве начального приближения выбирается одна из границ отрезка локализации, в которой совпадают знаки функции и ее второй производной.

Далее с помощью оператора *while* организован «бесконечный» цикл. Единица после слова *while* соответствует логическому значению «истина». В теле цикла происходит вычисление очередного приближения по формуле Ньютона. Цикл прекращается, если выполняется условие достижения заданной точности или же в случае превышения максимально допустимого числа итераций.

Для прекращения цикла используется оператор *break*. Так, с помощью операторов *while* и *break*, организован аналог цикла с постусловием. Если количество итераций превысило максимально допустимое, то с помощью функции *error* будет выведено соответствующее сообщение. Далее происходит формирование вектора *ans*, содержащего приближенное значение и количество итераций.

Обратите внимание, что здесь для вычисления производных от функции $f(x)$ используется встроенный в Mathcad оператор дифференцирования. Этот простой пример демонстрирует возможность использования встроенных вычислительных операторов (таких, как дифференцирование, интегрирование, суммирование и т.п.) при написании программных блоков.

Пример 3.

Пример 3. Найти решение уравнения $2x^3 - \cos(x) = 0$ методом касательных при помощи средств MathCAD с точностью $\varepsilon = 10^{-3}$.

Решение двумя способами представлено на рис. 16-17.

Постановка задачи Решить уравнение $2x^3 - \cos x = 0$

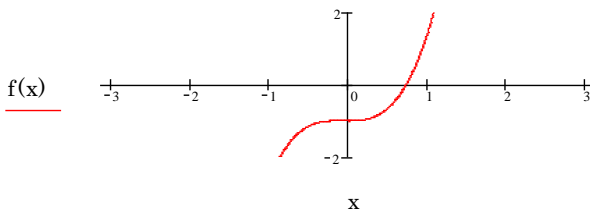
методом Ньютона с точностью $\epsilon = 10^{-3}$

Решение 1

(пошаговое, без использования встроенных функций MathCAD)

Отделим корни уравнения графически

$$f(x) := 2x^3 - \cos(x)$$



из графика следует, что один корень есть на отрезке $[a; b] = [0; 1]$

проверим условие Фурье $f(x_0) \cdot f''(x_0) > 0$, приняв за начальное

приближение конец интервала $x_0 = 1$

$$x := 0, 0.01.. 1$$

$$x_0 := 1$$

$$f(x_0) = 1.4597 \quad d^2f(x) := \frac{d^2}{dx^2} f(x) \quad \text{Взяли вторую производную} \\ f''(x)$$

$$f(x_0) \cdot d^2f(x_0) = 18.305$$

$f(x_0) \cdot f''(x_0) > 0$ Условие Фурье выполнено, значит итерационный процесс сходит

2.

Вычислим первое приближение

$$df(x) := \frac{d}{dx} f(x) \quad x_1 := x_0 - \frac{f(x_0)}{df(x_0)} \quad x_1 = 0.7866$$

Рис. 16

зададим точность вычислений

$$\varepsilon := 0.001$$

3. Проверим условие окончания итерационного процесса $|x_i - x_{i-1}| < \varepsilon$
 $|x_1 - x_0| = 0.2134$
 $0.2134 > \varepsilon$ условие нарушено
4. найдем второе приближение корня

$$x_2 := x_1 - \frac{f(x_1)}{df(x_1)} \quad x_2 = 0.7262$$
5. найдем третье приближение корня
 $|x_2 - x_1| = 0.0605$
 $0.0605 > \varepsilon$ условие нарушено

$$x_3 := x_2 - \frac{f(x_2)}{df(x_2)} \quad x_3 = 0.7214$$
6. найдем четвертое приближение корня
 $|x_3 - x_2| = 0.0047$
 $0.0047 > \varepsilon$ условие нарушено

$$x_4 := x_3 - \frac{f(x_3)}{df(x_3)} \quad x_4 = 0.7214$$

 $|x_4 - x_3| = 0.000028$

Условие выполнено: $0.000028 < 0.001$

Ответ: $x=0.7214$

Решение 2

(С использованием встроенных функций MathCAD)

$$\begin{array}{l}
 n := 100 \quad \text{приблизительное число итераций} \\
 i := 1..n \\
 x_{i+1} := \text{until} \left(|x_i - x_{i-1}| - \varepsilon, x_i - \frac{f(x_i)}{df(x_i)} \right) \quad \begin{array}{l} \text{реализация} \\ \text{итерационного} \\ \text{процесса с} \\ \text{использованием} \\ \text{встроенной функции until} \end{array} \\
 j := \text{last}(x) \quad \begin{array}{l} \text{определение числа} \\ \text{итераций, за которые} \\ \text{итерационный} \\ \text{процесс сошелся} \end{array} \\
 j = 5 \quad \begin{array}{l} \text{итерационная} \\ \text{последовательность} \end{array} \\
 x_{j-1} = 0.7214 \quad x = \begin{pmatrix} 1 \\ 0.7866 \\ 0.7262 \\ 0.7214 \\ 0.7214 \\ 0 \end{pmatrix} \\
 \text{корень уравнения}
 \end{array}$$

Рис. 17

Вопросы для самоконтроля для самоконтроля:

1. Что принимается за начальное приближение в методе Ньютона?
2. Каково условие окончания итерационного процесса?
3. Какие функции в системе Mathcad организуют итерационный процесс при решении методом Ньютона?
4. Как отыскать первое приближение корня в методе половинного деления?
5. Каково условие прекращения итерационного процесса в методе половинного деления?
6. Каковы недостатки метода половинного деления?

Лабораторная работа 1

Численные методы решения уравнений при помощи Mathcad.
Метод Ньютона, метод половинного деления

Цель работы: изучить методы приближенного решения уравнений- метод Ньютона, метод половинного деления, а также функции системы Mathcad, реализующие этот метод.

Задание

Решить уравнения методами половинного деления, методом хорд и методом Ньютона с точностью до 0.001 при помощи Mathcad.

Таблица 2

Но мер вари- анта	Задание 1	Задание 2	Задание 3
----------------------------	-----------	-----------	-----------

- | | | | |
|-----------|---------------------------------|---|-----------------------|
| 1 | $x^4 - x - 1 = 0$ | $x - \sin x = \frac{1}{4}$ | $2x - \lg x = 7$ |
| 2 | $2x^4 - x^2 - 10 = 0$ | $\operatorname{tg}(0.58x + 0.1) =$ | $3x - e^x = 0$ |
| 3 | $x^4 - 18x^2 + 6 = 0$ | $x \cdot 2^x + 6 = 1$ | $x = \sqrt{\lg(x+2)}$ |
| 4 | $x^4 + 4x^3 - 8x^2 - 17 = 0$ | $x - \cos x = 0$ | $2x + \lg(2x+3) =$ |
| 5 | $3x^4 + 4x^3 - 12x^2 + 1 = 0$ | $x^2 = \sin x$ | $x = (x+1)^3$ |
| 6 | $2x^4 - 8x^3 + 8x^2 - 1 = 0$ | $x^3 = \sin x$ | $x + \lg(x+1) = 1.1$ |
| 7 | $x^4 - 4x^3 - 8x^2 + 1 = 0$ | $\operatorname{ctgx} - \frac{x}{3} = 0$ | $x(x+1)^2 = 1$ |
| 8 | $2x^3 - 9x^2 - 60x + 1 = 0$ | $\operatorname{tg}(0.44x + 0.3) =$ | $2x + \lg x = -0.5$ |
| 9 | $x^4 - x^3 - 2x^2 + 3x - 3 = 0$ | $3x - \cos x - 1 = 0$ | $5x - 8 \ln x = 8$ |
| 10 | $3x^4 + 4x^3 - 12x^2 - 5 = 0$ | $x \cdot \lg x - \frac{6}{5} = 0$ | $x + \cos x = 1$ |
| 11 | $3x^4 - 8x^3 - 18x^2 + 2 = 0$ | $\operatorname{tg}(0.47x + 0.2) =$ | $x \cdot 2^x = 1$ |
| 12 | $6x^4 + x^3 - 2x^2 + 4 = 0$ | $x^2 - 8 \sin x = 0$ | $x - \cos x = 0$ |

13	$x^5 - x^2 - 60x + x - 8 = 0$	$2x - \lg x - 7 = 0$	$\sqrt{x+1} = \frac{1}{x}$
14	$x^3 - 3x^2 - 24x + 10 = 0$	$(x-1) \cdot 2^x = 1$	$\lg(x+2) + 2x =$
15	$2x^3 - 3x^2 - 12x + 1 = 0$	$x + \lg x = \frac{1}{2}$	$2 - x = \ln x$

Указания к работе

Перед выполнением работы внимательно изучите теоретический материал и ответьте на вопросы, просмотрите пример выполненного в системе MathCAD задания. Оформите работу так, как это сделано в контрольном примере.

ГЛАВА 3

РЕШЕНИЕ СИСТЕМ ЛИНЕЙНЫХ УРАВНЕНИЙ

Численное решение систем линейных алгебраических уравнений (СЛАУ) – одна из наиболее часто встречающихся задач в научно-технических исследованиях, математической физике (численное решение дифференциальных и интегральных уравнений), экономике, статистике. Поэтому, методам решения линейных алгебраических уравнений в современной вычислительной математике уделяется большое внимание.

Все методы решения СЛАУ делятся на две группы – точные (прямые) и итерационные. **Точные методы** позволяют получить решение системы линейных уравнений за конечное число арифметических операций (*метод Гаусса, метод квадратного корня, правило Крамера* и т. д.). Использование итерационных методов дает возможность найти приближенное решение системы с заданной степенью точности (метод простой итерации, метод Зейделя, метод последовательной релаксации).

При решении СЛАУ возникает необходимость выбора того или иного метода, который позволит получить эффективный результат с использованием вычислительной техники. В этой ситуации актуализируется проблема сравнительного анализа прямых и итерационных методов решения СЛАУ критериями сравнения точных и итерационных методов решения СЛАУ с использованием вычислительной техники будут:

- область применения метода;
- временные затраты на решение;
- погрешность результата.

Так, многие точные методы носят универсальный характер (схема Гаусса, схема Жордана и т.д.), и применяются для невырожденных систем общего вида, а некоторые— разработаны для специфических систем (точные клеточные методы решения больших систем линейных уравнений), что накладывает определенные ограничения на их использование.

Известно, что прикладные задачи часто требуют решения больших и сверхбольших СЛАУ с числом неизвестных более 1000. К таким СЛАУ, например, приводит численное решение двумерных и особенно трехмерных задач математической физики, в которых условия физической и геометрической аппроксимации двумерной и трехмерной области диктуют использование достаточно мелкой расчетной сетки с большим числом расчетных узлов по линейному размеру. Решения задач СЛАУ большой размерности целесообразно осуществлять итерационными методами, так как использование прямых методов невозможно из-за ограничений доступной оперативной памяти ЭВМ, а также из-за необходимости выполнения чрезмерно большого числа арифметических операций.

3.1 Понятия и определения. Прямые методы решения СЛАУ

Определение. Система m линейных уравнений с n неизвестными (СЛУ) x_1, x_2, \dots, x_n имеет вид:

$$\begin{cases} a_{11}x_1 + a_{12}x_2 + \dots + a_{1n}x_n = c_1; \\ a_{21}x_1 + a_{22}x_2 + \dots + a_{2n}x_n = c_2; \\ \dots\dots\dots \\ a_{m1}x_1 + a_{m2}x_2 + \dots + a_{mn}x_n = c_m. \end{cases} \quad (3.1)$$

Здесь a_{ij} , c_i ($i = 1, 2, \dots, m$; $j = 1, 2, \dots, n$) - заданные числа, причём числа a_{ij} называются *коэффициентами при неизвестных*; c_i - *свободными членами*.

Введём матрицы:

$$A = \begin{pmatrix} a_{11} & a_{12} & \dots & a_{1n} \\ a_{21} & a_{22} & \dots & a_{2n} \\ \dots & \dots & \dots & \dots \\ a_{m1} & a_{m2} & \dots & a_{mn} \end{pmatrix}; \quad X = \begin{pmatrix} x_1 \\ x_2 \\ \dots \\ x_n \end{pmatrix}; \quad C = \begin{pmatrix} c_1 \\ c_2 \\ \dots \\ c_m \end{pmatrix}.$$

Здесь A , C - заданные матрицы; X - неизвестный вектор-столбец.

Определение. Матрица A называется *матрицей коэффициентов* при неизвестных или *основной матрицей системы*, матрица C - *вектор-столбцом свободных коэффициентов*.

Определение. Определитель основной матрицы системы $\det A = \Delta$ называется *определителем системы* (3.1).

Определение. *Решением системы* (3.1) называется такая совокупность n чисел $x_1 = k_1, x_2 = k_2, \dots, x_n = k_n$, при подстановке которых каждое уравнение системы (1) обращается в верное тождество.

Определение. Две системы уравнений называются *равносильными*, или *эквивалентными*, если они имеют одно и то же множество решений.

Эквивалентность матриц A и B обозначается: $A \sim B$.

Определение. Система называется *совместной*, если она имеет хотя бы одно решение, и *несовместной*, если она не имеет решений.

Определение. Совместная система уравнений называется *определённой*, если она имеет единственное решение, и *неопределённой*, если она имеет более одного решения.

3.2 Решение систем линейных уравнений матричным способом

Рассмотрим систему n линейных уравнений с n неизвестными:

$$\begin{cases} a_{11}x_1 + a_{12}x_2 + \dots + a_{1n}x_n = c_1; \\ a_{21}x_1 + a_{22}x_2 + \dots + a_{2n}x_n = c_2; \\ \dots\dots\dots \\ a_{n1}x_1 + a_{n2}x_2 + \dots + a_{nn}x_n = c_n. \end{cases} \quad (3.2)$$

Это частный случай СЛУ (1), когда число уравнений m равно числу неизвестных n . Заметим, что только при выполнении этого условия применим матричный метод.

По правилу умножения матриц систему (2) можно записать в виде матричного уравнения

$$A \cdot X = C, \quad (3.3)$$

где A - заданная матрица (основная матрица системы);

C - заданный вектор-столбец свободных коэффициентов;

X - неизвестный вектор-столбец.

Решением матричного уравнения (3) является такой вектор-столбец X , который обращает уравнение (3) в тождество.

$$X = A^{-1} \cdot C. \quad (3.4)$$

Поскольку обратную матрицу можно найти только при условии, что $\det A = \Delta \neq 0$, то и сам матричный способ решения системы линейных уравнений можно применять при выполнении этого условия.

Пример. Решите систему линейных алгебраических уравнений матричным способом

$$\begin{cases} 3x_1 - x_2 = 5, \\ -2x_1 + x_2 + x_3 = 0, \\ 2x_1 - x_2 + 4x_3 = 15 \end{cases}$$

при помощи инструментов Mathcad .

Решение систем линейных уравнений точными методами

Матричный метод

Решите СЛУ

$$\begin{cases} 3 \cdot x_1 - x_2 = 5, \\ -2 \cdot x_1 + x_2 + x_3 = 0, \\ 2 \cdot x_1 - x_2 + 4 \cdot x_3 = 15. \end{cases}$$

Матрица системы

$$A := \begin{pmatrix} 3 & -1 & 0 \\ -2 & 1 & 1 \\ 2 & -1 & 4 \end{pmatrix}$$

Матрица правой части

$$B := \begin{pmatrix} 5 \\ 0 \\ 15 \end{pmatrix}$$

Вычисление определителя

$$|A| = 5$$

Определитель отличен от нуля,
система имеет единственное
решение

Вычисление решения системы

$$X := A^{-1} \cdot B$$

$$X = \begin{pmatrix} 2 \\ 1 \\ 3 \end{pmatrix}$$

Решение системы с помощью
функции *lsolve*

$$X := \text{lsolve}(A, B)$$

$$X = \begin{pmatrix} 2 \\ 1 \\ 3 \end{pmatrix}$$

Проверка правильности
решения системы

$$A \cdot X - B = \begin{pmatrix} 0 \\ 0 \\ 0 \end{pmatrix}$$

Рис. 18

3.3 Решение систем линейных уравнений методом Крамера

Метод Крамера, как и матричный способ, применяется для решения систем n линейных уравнений с n неизвестными при условии, что основной определитель системы не равен нулю:

$$\det A = \Delta \neq 0.$$

Рассмотрим систему n линейных уравнений с n неизвестными вида (3.2).

Пусть A - основная матрица системы (матрица, составленная из коэффициентов при неизвестных).

Рассмотрим определители:

$$\Delta = \begin{vmatrix} a_{11} & a_{12} & \dots & a_{1n} \\ a_{21} & a_{22} & \dots & a_{2n} \\ \dots & \dots & \dots & \dots \\ a_{n1} & a_{n2} & \dots & a_{nn} \end{vmatrix}; \quad \Delta_{x_1} = \begin{vmatrix} c_1 & a_{12} & \dots & a_{1n} \\ c_2 & a_{22} & \dots & a_{2n} \\ \dots & \dots & \dots & \dots \\ c_n & a_{n2} & \dots & a_{nn} \end{vmatrix};$$

$$\Delta_{x_2} = \begin{vmatrix} a_{11} & c_1 & \dots & a_{1n} \\ a_{21} & c_2 & \dots & a_{2n} \\ \dots & \dots & \dots & \dots \\ a_{n1} & c_n & \dots & a_{nn} \end{vmatrix}; \quad \dots; \quad \Delta_{x_n} = \begin{vmatrix} a_{11} & a_{12} & \dots & c_1 \\ a_{21} & a_{22} & \dots & c_2 \\ \dots & \dots & \dots & \dots \\ a_{n1} & a_{n2} & \dots & c_n \end{vmatrix}.$$

Определители $\Delta_{x_1}, \Delta_{x_2}, \dots, \Delta_{x_n}$ получаются из определителя системы Δ заменой свободными членами элементов соответственно первого, второго, ..., n -го столбцов.

Если $\det A = \Delta \neq 0$, то существует, и притом единственное, решение этой системы, которое вычисляется по *формулам Крамера*:

$$x_1 = \frac{\Delta_{x_1}}{\Delta}, \quad x_2 = \frac{\Delta_{x_2}}{\Delta}, \quad \dots, \quad x_n = \frac{\Delta_{x_n}}{\Delta}.$$

При $\det A = \Delta = \Delta_{x_i} = 0, i = \overline{1, n}$ система вида (3.2) имеет бесконечное множество решений.

В случае, если $\det A = \Delta = 0$ и хотя бы один $\Delta_{x_i} \neq 0, i = \overline{1, n}$ система не имеет решений.

Пример решения системы линейных уравнений с тремя переменными методом Крамера в универсальной математической системе MathCAD представлен ниже на рис. 19.

Решение систем линейных уравнений методом Крамер

$$\text{ORIGIN}:=1 \quad \begin{cases} x_1 + 2x_2 + 3x_3 = 7 \\ x_1 - 3x_2 + 2x_3 = 5 \\ x_1 + x_2 + x_3 = 3 \end{cases}$$

$$A := \begin{pmatrix} 1 & 2 & 3 \\ 1 & -3 & 2 \\ 1 & 1 & 1 \end{pmatrix}$$

$$b := \begin{pmatrix} 7 \\ 5 \\ 3 \end{pmatrix}$$

$$|A| = 9$$

Система имеет единственное решение, т.к. главный определитель не равен нулю

$$A1 := \begin{pmatrix} 7 & 2 & 3 \\ 5 & -3 & 2 \\ 3 & 1 & 1 \end{pmatrix}$$

$$|A1| = 9$$

$$A3 := \begin{pmatrix} 1 & 2 & 7 \\ 1 & -3 & 5 \\ 1 & 1 & 3 \end{pmatrix}$$

$$|A3| = 18 \quad A2 := \begin{pmatrix} 1 & 7 & 3 \\ 1 & 5 & 2 \\ 1 & 3 & 1 \end{pmatrix}$$

$$|A2| = 0$$

$$x1 := \frac{|A1|}{|A|}$$

$$x1 = 1$$

$$x2 := \frac{|A2|}{|A|}$$

$$x2 = 0$$

$$x3 := \frac{|A3|}{|A|}$$

$$x3 = 2$$

Проверка

$$x1 + 2 \cdot x2 + 3 \cdot x3 = 7$$

$$x1 - 3 \cdot x2 + 2 \cdot x3 = 5$$

$$x1 + x2 + x3 = 3$$

Рис. 19

3.4 Решение систем линейных уравнений методом Гаусса

Метод Гаусса является наиболее распространённым методом решения систем линейных уравнений как вида (3.22), так и произвольных систем m линейных уравнений с n неизвестными вида (3.1).

Метод Гаусса - это метод последовательного исключения неизвестных. Исключение неизвестных осуществляется при помощи следующих преобразований системы:

- умножения уравнения системы на число, отличное от нуля;
- перестановки местами двух уравнений системы;
- прибавления к одному уравнению системы другого, умноженного на какое-либо число.

Процесс исключения неизвестных состоит в переводе исходной системы линейных уравнений (1) в такую равносильную систему, в которой каждое следующее уравнение содержит, по крайней мере, одним неизвестным меньше, чем предыдущее. Если этот процесс завершён, то решение системы находится следующим образом: из последнего уравнения определяется одно из неизвестных, затем подстановкой в предыдущее уравнение определяется другое неизвестное, далее вновь подстановкой в предшествующее уравнение находится ещё одно неизвестное и т.д., пока не определяются все неизвестные. Пусть в системе уравнений

$$\begin{cases} a_{11}x_1 + a_{12}x_2 + \dots + a_{1n}x_n = c_1; \\ a_{21}x_1 + a_{22}x_2 + \dots + a_{2n}x_n = c_2; \\ \dots\dots\dots \\ a_{m1}x_1 + a_{m2}x_2 + \dots + a_{mn}x_n = c_m. \end{cases}$$

первый элемент $a_{11} \neq 0$. Назовем его ведущим элементом первой строки. Поделим все элементы этой строки на a_{11} и исключим x_1 из всех последующих строк, начиная со второй, путем вычитания первой (преобразованной), умноженной на коэффициент при x_1 в соответствующей строке. Получим

Решение системы линейных уравнений точными методами

Метод Гаусса

**1. Количество уравнений m равно числу переменных n
m=n**

Решите систему
$$\begin{cases} x_1 + 2 \cdot x_2 + 3 \cdot x_3 = 7, \\ x_1 - 3 \cdot x_2 + 2 \cdot x_3 = 5, \\ x_1 + x_2 + x_3 = 3. \end{cases}$$

ORIGIN:=1

задание нумерации строк и столбцов матрицы с един

Матрица системы

Матрица правой
части

$$A := \begin{pmatrix} 1 & 2 & 3 \\ 1 & -3 & 2 \\ 1 & 1 & 1 \end{pmatrix}$$

$$B := \begin{pmatrix} 7 \\ 5 \\ 3 \end{pmatrix}$$

Сформируем расширенную матрицу системы Ar,
добавлением к матрице системы матрицы правых частей

$$Ar = \begin{pmatrix} 1 & 2 & 3 & 7 \\ 1 & -3 & 2 & 5 \\ 1 & 1 & 1 & 3 \end{pmatrix}$$

Ar := augment(A, B)

Приведем расширенную матрицу Ar к ступенчатому виду
при помощи функции rref. Образованную матрицу
обозначим Ag

Ag := rref(Ar)

Примечание Функция rref выполняет
прямой
и обратный ход Гауссова
преобразования

$$Ag = \begin{pmatrix} 1 & 0 & 0 & 1 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 2 \end{pmatrix}$$

Выделим четвертый столбец решений матрицы Ag при
помощи функции submatrix

X := submatrix(Ag, 1, 3, 4, 4)

$$X = \begin{pmatrix} 1 \\ 0 \\ 2 \end{pmatrix}$$

Проверка правильности
системы

$$A \cdot X - B = \begin{pmatrix} 0 \\ 0 \\ 0 \end{pmatrix}$$

Рис. 20

Пример несовместной системы

Решите систему

ORIGIN := 1

$$\begin{cases} x_1 + x_2 = 1, \\ x_1 + x_2 + x_3 = 4, \\ x_2 + x_3 + x_4 + x_5 = 2, \\ 2 \cdot x_1 + 3 \cdot x_2 + 2 \cdot x_3 + x_4 + x_5 = 7, \\ 3 \cdot x_1 + 3x_2 + 2 \cdot x_3 = 10. \end{cases}$$

$$A := \begin{pmatrix} 1 & 1 & 0 & 0 & 0 \\ 1 & 1 & 1 & 0 & 0 \\ 0 & 1 & 1 & 1 & 1 \\ 2 & 3 & 2 & 1 & 1 \\ 3 & 3 & 2 & 0 & 0 \end{pmatrix} \quad B := \begin{pmatrix} 1 \\ 4 \\ 2 \\ 7 \\ 10 \end{pmatrix}$$

Ar := augment(A,B)

$$Ar = \begin{pmatrix} 1 & 1 & 0 & 0 & 0 & 1 \\ 1 & 1 & 1 & 0 & 0 & 4 \\ 0 & 1 & 1 & 1 & 1 & 2 \\ 2 & 3 & 2 & 1 & 1 & 7 \\ 3 & 3 & 2 & 0 & 0 & 10 \end{pmatrix}$$

$$\text{rank}(A) = 3 \quad \text{rank}(Ar) = 4$$

Ранги матриц A и A1 не равны, поэтому система **НСОВМЕСТНА**.

Рис. 21

3. Количество уравнений m не равно числу переменных n

Пример совместной системы

Решите систему

$$\begin{cases} x_1 + x_2 = 1, \\ x_1 + x_2 + x_3 = 4, \\ x_2 + x_3 + x_4 + x_5 = 2, \\ 2 \cdot x_1 + 3 \cdot x_2 + 2 \cdot x_3 + x_4 + x_5 = 7, \\ 3 \cdot x_1 + 3x_2 + 2 \cdot x_3 = 9. \end{cases}$$

$$A := \begin{pmatrix} 1 & 1 & 0 & 0 & 0 \\ 1 & 1 & 1 & 0 & 0 \\ 0 & 1 & 1 & 1 & 1 \\ 2 & 3 & 2 & 1 & 1 \\ 3 & 3 & 2 & 0 & 0 \end{pmatrix}$$

$$B := \begin{pmatrix} 1 \\ 4 \\ 2 \\ 7 \\ 9 \end{pmatrix}$$

$$Ar := \text{augment}(A, B)$$

$$Ar = \begin{pmatrix} 1 & 1 & 0 & 0 & 0 & 1 \\ 1 & 1 & 1 & 0 & 0 & 4 \\ 0 & 1 & 1 & 1 & 1 & 2 \\ 2 & 3 & 2 & 1 & 1 & 7 \\ 3 & 3 & 2 & 0 & 0 & 9 \end{pmatrix}$$

$$\text{rank}(A) = 3$$

$$\text{rank}(Ar) = 3$$

Ранги матриц A и Ar равны, поэтому система совместна.

Приведем матрицу к ступенчатому виду

$$AG := \text{rref}(Ar)$$
$$AG = \begin{pmatrix} 1 & 0 & 0 & -1 & -1 & 2 \\ 0 & 1 & 0 & 1 & 1 & -1 \\ 0 & 0 & 1 & 0 & 0 & 3 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{pmatrix}$$

Рис. 22

Вычислим базисный минор при переменных x_1, x_2, x_3

$$A_3 := \left| \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix} \right| \quad A_3 = 1$$

Базисный минор A_3 при переменных x_1, x_2, x_3 равен 1 (не равен нулю), поэтому эти переменные являются основными, а переменные x_4, x_5 - свободными переменными

Запишем эквивалентную систему и решим ее относительно базисных переменных x_1, x_2, x_3

$$x_4 := 0 \quad x_5 := 0 \quad \text{Инициализация переменных}$$

Given Блок решений системы

$$x_1 - x_4 - x_5 = 2$$

$$x_2 + x_4 + x_5 = -1$$

$$x_3 = 3$$

$$\text{Find}(x_1, x_2, x_3, x_4, x_5) \rightarrow \begin{pmatrix} x_4 + x_5 + 2 \\ -x_4 - x_5 - 1 \\ 3 \\ x_4 \\ x_5 \end{pmatrix} \quad \text{Найдено общее решение системы}$$

$$X(x) := \begin{pmatrix} x_4 + x_5 + 2 \\ -x_4 - x_5 - 1 \\ 3 \\ x_4 \\ x_5 \end{pmatrix} \quad \text{Запишем общее решение в виде функции}$$

Выполним проверку решения, подставив в уравнение $AX=B$ найденное общее решение

$$A \cdot X(x) - B \rightarrow \begin{pmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{pmatrix}$$

Рис. 23

Лабораторная работа 2

Точные методы решения систем линейных уравнений при помощи Mathcad.

Цель работы: изучить методы точного решения уравнений: метод Крамера, матричный метод, метод Гаусса.

Задание 1. Решите систему линейных уравнений матричным способом.

Ответ округлите до 10^{-4} .

Задание 2. Решите систему линейных уравнений методами Крамера и Гаусса (с точностью до 10^{-3}).

Указания к выполнению

Встроенная функция **lsolve** (A , B) возвращает (находит) решение системы линейных уравнений, основные коэффициенты которой определены в матрице A , свободные коэффициенты – в векторе B . Реализация метода Гаусса в Mathcad может быть следующей:

- введите **ORIGIN := 1** нумерация строк и столбцов матрицы начинается с 1;
- определите матрицы A и B (они уже определены в примере);
- сформируйте расширенную матрицу, используя встроенную функцию **augment** (A , B):
 - $Ar := \text{augment}(A, B)$ - формирует матрицу, в первых столбцах которой содержится матрица A , а в последних – матрица B . Далее, проверяем правильность сформированной матрицы:
 - $Ar =$
 - приведите расширенную матрицу Ar к ступенчатому виду с единичной матрицей в первых столбцах, т.е. выполните прямой и обратные ходы метода Гаусса, используя встроенную функцию **gref** (Ar):
 - $Ag := \text{gref}(Ar)$.

- Далее, проверяем правильность сформированной матрицы:
- $Ag =$
- сформируйте столбец решения системы, используя встроенную функцию `submatrix (A, ir, jr, ic, jc)`:
- $X := \text{submatrix}(Ag, 1, 3, 5, 5)$

Функция `submatrix (A, ir, jr, ic, jc)` формирует матрицу, которая является блоком матрицы A , расположенным в строках с ***ir*** по ***jr*** и в столбцах с ***ic*** по ***jc***. После чего делаем проверку системы, АХ-В. Если все выполнено верно, должны получить нуль.

Используем также встроенную функцию `lsolve(A,B)` для решения данной системы.

Задания к лабораторной работе представлены в таблице 3

Таблица 3

Вариант	Задание 1	Задание 2
1	$\begin{cases} 4.3x_1 - 2.9x_2 - 1.3x_3 + 5.2x_4 = 8.3, \\ 5.6x_1 - 8.3x_2 + 9.2x_3 - 3.5x_4 = 4.5, \\ 6.3x_1 + 8.6x_2 - 2.1x_3 + 8.9x_4 = 8.1, \\ 7.9x_1 - 7.5x_2 + 4.5x_3 + 7.2x_4 = 5.7. \end{cases}$	$\begin{cases} 0.51x - 1.9y - 0.3z = 1.1, \\ 1.9x + 3.2y - 0.5z = 1.2, \\ 3.6x + 0.3y - 1.4z = 5.2. \end{cases}$
2	$\begin{cases} 0.7x_1 - 0.3x_2 + 0.8x_3 + 0.1x_4 = 0.4, \\ 0.9x_1 + 1.4x_2 - 0.4x_3 - 0.4x_4 = 0.2, \\ 0.8x_1 + 1.6x_2 - 0.8x_3 + 0.7x_4 = 0.9, \\ 0.2x_1 - 0.4x_2 - 0.2x_3 - 0.4x_4 = 0.5. \end{cases}$	$\begin{cases} 0.51x - 0.29y + 0.13z = 1.4, \\ 0.95x - 1.32y + 1.43z = 4.9, \\ 0.71x - 0.94y + 0.26z = 1.7. \end{cases}$
3	$\begin{cases} 4.1x_1 - 3.2x_2 + 5.1x_3 + 5.8x_4 = 3.9, \\ 6.6x_1 - 8.4x_2 + 8.2x_3 - 9.9x_4 = 7.2, \\ 9.3x_1 - 7.6x_2 + 9.1x_3 - 3.8x_4 = 8.1, \\ 7.9x_1 - 5.5x_2 + 4.2x_3 - 4.7x_4 = 9.1. \end{cases}$	$\begin{cases} 9.24x + 5.29y - 7.21z = 5.9, \\ 0.96x - 4.42y + 3.69z = 8.6, \\ 3.71x - 7.47y - 5.22z = 9.4. \end{cases}$

$$\begin{array}{l}
4 \quad \begin{cases} 2.6x_1 - 3.1x_2 - 2.7x_3 - 2.5x_4 = 1.6, \\ 4.6x_1 - 1.4x_2 - 1.2x_3 - 3.7x_4 = -4.8, \\ 1.3x_1 - 6.6x_2 - 4.1x_3 - 1.6x_4 = -2.2, \\ 3.9x_1 + 2.7x_2 - 2.8x_3 - 1.3x_4 = 0.3. \end{cases} \quad \begin{cases} -0.3x - 0.9y + 0.3z = 1.61, \\ -0.5x + 0.2y - 0.8z = 1.59, \\ -0.1x - 0.7y + 0.2z = 2.71. \end{cases} \\
5 \quad \begin{cases} 7.8x_1 + 9.5x_2 - 3.7x_3 - 9.3x_4 = 7.4, \\ 4.9x_1 - 4.8x_2 - 6.9x_3 - 7.6x_4 = -4.9, \\ 7.6x_1 - 3.6x_2 - 9.1x_3 + 6.6x_4 = 6.2, \\ 8.4x_1 + 4.7x_2 - 2.8x_3 - 1.3x_4 = 4.3. \end{cases} \quad \begin{cases} 7.8x + 5.6y - 2.6z = 8.12, \\ 8.5x - 6.9y + 5.8z = 9.5, \\ 9.4x + 0.6y - 2.7z = 5.13. \end{cases} \\
6 \quad \begin{cases} 0.6x_1 - 0.1x_2 - 0.7x_3 - 0.5x_4 = 1.8, \\ 4.9x_1 - 7.4x_2 - 8.2x_3 + 9.2x_4 = 3.9, \\ 1.4x_1 + 1.6x_2 + 1.1x_3 - 1.3x_4 = 5.9, \\ 8.2x_1 + 9.7x_2 - 6.8x_3 + 7.9x_4 = 4.6. \end{cases} \quad \begin{cases} 5.3x - 6.3y - 6.3z = 8.3, \\ 8.5x + 5.2y - 9.8z = 4.5, \\ 7.3x + 3.7y + 5.8z = 9.1. \end{cases} \\
7 \quad \begin{cases} 2.6x_1 - 5.1x_2 + 0.7x_3 - 0.5x_4 = 2.8, \\ 4.9x_1 - 7.4x_2 - 8.2x_3 + 9.2x_4 = 3.9, \\ 5.4x_1 + 1.6x_2 + 1.1x_3 - 1.3x_4 = 3.9, \\ 8.2x_1 + 6.7x_2 - 6.8x_3 + 7.9x_4 = 2.6. \end{cases} \quad \begin{cases} 1.6x - 0.6y + 0.9z = 1.6, \\ 1.9x - 0.9y - 0.4z = 1.3, \\ 2.3x + 0.6y + 0.7z = 2.2. \end{cases} \\
8 \quad \begin{cases} 2.6x_1 - 3.1x_2 - 2.7x_3 - 2.5x_4 = 1.6, \\ 4.6x_1 - 1.4x_2 - 1.2x_3 - 3.7x_4 = -4.8, \\ 1.3x_1 - 6.6x_2 - 4.1x_3 - 1.6x_4 = -2.2, \\ 3.9x_1 + 2.7x_2 - 2.8x_3 - 1.3x_4 = 0.3. \end{cases} \quad \begin{cases} 2.6x - 7.9y - 8.3z = 2.11, \\ -3.5x + 2.2y - 3.8z = 3.57, \\ 5.1x + 7.7y - 6.2z = 6.93. \end{cases} \\
9 \quad \begin{cases} 0.7x_1 - 0.3x_2 + 0.8x_3 + 0.1x_4 = 0.4, \\ 0.9x_1 + 1.4x_2 - 0.4x_3 - 0.4x_4 = 0.2, \\ 0.8x_1 + 1.6x_2 - 0.8x_3 + 0.7x_4 = 0.9, \\ 0.2x_1 - 0.4x_2 - 0.2x_3 - 0.4x_4 = 0.5. \end{cases} \quad \begin{cases} 0.51x - 0.29y + 0.13z = 1.4, \\ 0.95x - 1.32y + 1.43z = 4.9, \\ 0.71x - 0.94y + 0.26z = 1.7. \end{cases} \\
10 \quad \begin{cases} 4.1x_1 - 3.2x_2 + 5.1x_3 + 5.8x_4 = 3.9, \\ 6.6x_1 - 8.4x_2 + 8.2x_3 - 9.9x_4 = 7.2, \\ 9.3x_1 - 7.6x_2 + 9.1x_3 - 3.8x_4 = 8.1, \\ 7.9x_1 - 5.5x_2 + 4.2x_3 - 4.7x_4 = 9.1. \end{cases} \quad \begin{cases} 9.24x + 5.29y - 7.21z = 5.9, \\ 0.96x - 4.42y + 3.69z = 8.6, \\ 3.71x - 7.47y - 5.22z = 9.4. \end{cases} \\
11 \quad \begin{cases} 5.1x_1 - 3.2x_2 + 5.1x_3 + 5.8x_4 = 13.9, \\ 10.6x_1 - 8.4x_2 + 8.2x_3 - 9.9x_4 = 7.2, \\ 9.3x_1 - 7.6x_2 + 9.1x_3 - 3.8x_4 = 18.1, \\ 5.9x_1 - 5.5x_2 + 4.2x_3 - 4.7x_4 = 9.1. \end{cases} \quad \begin{cases} 9.24x + 5.29y - 7.21z = 5.9, \\ 2.96x + 2.42y + 3.69z = 1.6, \\ 4.71x - 7.47y - 5.22z = 0.4. \end{cases}
\end{array}$$

$$12 \quad \begin{cases} 1.8x_1 + 9.5x_2 - 3.7x_3 - 9.3x_4 = 7.4, \\ 0.9x_1 - 4.8x_2 - 6.9x_3 - 7.6x_4 = -4.9, \\ 7.6x_1 - 3.6x_2 - 9.1x_3 + 6.6x_4 = 1.2, \\ 8.4x_1 + 4.7x_2 - 2.8x_3 - 1.3x_4 = 4.3. \end{cases} \quad \begin{cases} 9.24x + 5.29y - 7.21z = 5.9, \\ 0.96x - 4.42y + 3.69z = 8.6, \\ 3.71x - 7.47y - 5.22z = 9.4. \end{cases}$$

$$13 \quad \begin{cases} 4.1x_1 - 3.2x_2 + 5.1x_3 + 5.8x_4 = 3.9, \\ 6.6x_1 - 8.4x_2 + 8.2x_3 - 9.9x_4 = 7.2, \\ 9.3x_1 - 7.6x_2 + 9.1x_3 - 3.8x_4 = 2.1, \\ 7.6x_1 + 5.5x_2 + 4.2x_3 - 4.7x_4 = 9.1. \end{cases} \quad \begin{cases} 6.2x + 5.29y - 7.21z = 5.9, \\ 1.96x - 4.42y + 3.69z = 0.6, \\ 3.71x - 7.47y - 5.22z = 9.4. \end{cases}$$

$$14 \quad \begin{cases} 8.7x_1 - 0.3x_2 + 0.8x_3 + 0.1x_4 = 0.4, \\ 0.9x_1 + 1.4x_2 - 0.4x_3 - 0.4x_4 = 1.2, \\ 7.8x_1 + 1.6x_2 - 0.8x_3 + 0.7x_4 = 0.9, \\ 6.2x_1 - 0.4x_2 - 0.2x_3 - 0.4x_4 = 0.5. \end{cases} \quad \begin{cases} 7.8x + 5.6y - 2.6z = 8.12, \\ 8.5x + 6.9y + 5.8z = 2.5, \\ 5.4x + 0.6y - 2.7z = 5.13. \end{cases}$$

$$15 \quad \begin{cases} 0.7x_1 - 0.3x_2 + 0.8x_3 + 0.1x_4 = 0.4, \\ 1.9x_1 + 1.4x_2 - 0.4x_3 - 0.4x_4 = 1.2, \\ 7.8x_1 + 6.6x_2 - 0.8x_3 + 0.7x_4 = 3.9, \\ 6.2x_1 - 0.4x_2 - 0.2x_3 - 0.4x_4 = 0.5. \end{cases} \quad \begin{cases} 4.8x + 5.6y - 2.6z = 0.12, \\ 9.5x + 6.9y + 5.8z = 6.5, \\ 1.4x + 4.6y - 2.7z = 5.13. \end{cases}$$

Вопросы для самоконтроля:

1. Как ввести матрицу в системе Mathcad?
2. Для чего нужны функции ORIGIN, rref, augment?
3. Когда можно применять матричный метод?
4. Когда можно решать систему методом Крамера?
5. Как решить СЛУ при условии неравенства числа уравнений числу переменных средствами Mathcad?
6. Какие переменные являются базисными, а какие - свободными?
7. Что такое базисный минор?
8. Какое решение системы является общим?
9. Как выполнить проверку СЛУ?

3.5 Приближенные методы решения систем линейных уравнений

3.5.1 Решение систем линейных уравнений методом итераций

Пусть дана линейная система

$$\begin{cases} a_{11}x_1 + a_{12}x_2 + \dots + a_{1n}x_n = b_1, \\ a_{21}x_1 + a_{22}x_2 + \dots + a_{2n}x_n = b_2, \\ \dots\dots\dots \\ a_{n1}x_1 + a_{n2}x_2 + \dots + a_{nn}x_n = b_n. \end{cases} \quad (3.5)$$

Как было показано выше, данную систему можно представить в виде матричного уравнения $A \cdot X = B$.

Предполагая, что диагональные коэффициенты $a_{ij} \neq 0, (i = \overline{1, n}, j = \overline{1, n})$ разрешим первое уравнение системы (3.5) относительно x_1 , второе - относительно x_2 и т. д.

Тогда получим эквивалентную систему (нормальный вид системы)

$$\begin{cases} x_1 = \beta_1 + \alpha_{12}x_2 + \alpha_{13}x_3 + \dots + \alpha_{1n}x_n, \\ x_2 = \beta_2 + \alpha_{21}x_1 + \alpha_{23}x_3 + \dots + \alpha_{2n}x_n, \\ \dots\dots\dots \\ x_n = \beta_n + \alpha_{n1}x_1 + \alpha_{n2}x_2 + \dots + \alpha_{n, n-1}x_{n-1}, \end{cases} \quad (3.6)$$

где $\beta_i = \frac{b_i}{a_{ii}}, \alpha_{ij} = -\frac{\alpha_{ij}}{a_{ii}},$ если $i \neq j$ и $\alpha_{ii} = 0$ при $i = j$.

Введем матрицы $A = \begin{pmatrix} \alpha_{11} & \alpha_{12} & \dots & \alpha_{1n} \\ \alpha_{21} & \alpha_{22} & \dots & \alpha_{2n} \\ \dots & \dots & \dots & \dots \\ \alpha_{n1} & \alpha_{n2} & \dots & \alpha_{nn} \end{pmatrix}$ и $\beta = \begin{pmatrix} \beta_1 \\ \beta_2 \\ \dots \\ \beta_n \end{pmatrix}$.

систему (3.6) можно записать в матричной форме $x = \beta + \alpha x$

Любое $(k + 1)$ -е приближение вычисляется по формуле

$$x^{(k+1)} = \beta + \alpha x^{(k)} \quad (3.7)$$

Напишем формулы приближений в развернутом виде:

$$\begin{cases} x_i^{(0)} = \beta_i, \\ x_i^{(k+1)} = \beta_i + \sum_{j=1}^n \alpha_{i,j} x_j^{(k)}, \\ \alpha_{ii} = 0, \quad i = 1, \dots, n; \quad k = 0, 1, 2, \dots \end{cases} \quad (3.8)$$

Приведем достаточное условие сходимости метода итераций.

Теорема. Процесс итерации для приведенной линейной системы (3.6) сходится к единственному ее решению, если какая-нибудь каноническая норма матрицы A меньше единицы, т.е. для итерационного процесса (3.8) достаточное условие есть $\|\alpha\| < 1$.

Следствие 1. Процесс итерации для системы (3.8) сходится, если:

$$\|\alpha\|_m = \max_i \sum_j |\alpha_{ij}| < 1$$

(m -норма или неопределенная норма)

$$\|\alpha\|_l = \max_j \sum_i |\alpha_{ij}| < 1$$

(l -норма или норма $L1$)

Или

$$\|\alpha\|_k = \sqrt{\sum_{ij} |\alpha_{ij}|^2} < 1$$

(k -норма или Евклидова норма).

Следствие 2. Для системы (3.8) процесс итерации сходится, если выполнены неравенства:

$$1. \quad |\alpha_{ii}| > \sum_{j=1}^n |\alpha_{ij}| \quad (i = 1, 2, \dots, n)$$

Или

$$|\alpha_{jj}| > \sum_{i=1}^n |\alpha_{ij}| \quad (j = 1, 2, \dots, n)$$

т.е. сходимость имеет место, если модули диагональных элементов матрицы A системы (3.5) или каждой строки превышают сумму модулей недиагональных элементов этой строки, или же для каждого столбца превышают сумму модулей недиагональных элементов этого столбца.

Пример нахождения норм матрицы

Пусть матрица A равна $A = \begin{bmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \\ 7 & 8 & 9 \end{bmatrix}$

Имеем:

$$\|A\|_m = \max(1 + 2 + 3, 4 + 5 + 6, 7 + 8 + 9) = \max(6, 15, 24) = 24$$

$$\|A\|_l = \max(1 + 4 + 7, 2 + 5 + 8, 3 + 6 + 9) = \max(12, 15, 18) = 18$$

$$\|A\|_k = \sqrt{1^2 + 2^2 + 3^2 + 4^2 + 5^2 + 6^2 + 7^2 + 8^2 + 9^2} = \sqrt{285} \approx 16,9$$

В Mathcad существуют специальные функции для вычисления норм матриц:

normi(A)

Возвращает неопределенную норму матрицы A .

norm1(A)

Возвращает $L1$, норму матрицы A .

norme(A)

Возвращает Евклидову норму матрицы A .

В качестве условия окончания итерационного процесса можно взять условие

$$\frac{\|x^{(k+1)} - x^{(k)}\|}{\|x^{(k+1)}\|} \leq \varepsilon,$$

ε - заданная погрешность приближенного решения $x \approx x^{(k+1)}$.

Пример .

Решить систему методом итераций.

$$\begin{cases} 100x_1 + 6x_2 - 2x_3 = 200, \\ 3x_1 + 200x_2 - 10x_3 = 600, \\ x_1 - 2x_2 - 100x_3 = 500 \end{cases}$$

Диагональные коэффициенты 100; 200; 100 системы значительно преобладают над остальными коэффициентами при неизвестных, т.е., выполняется следствие 2.

Приведем эту систему к нормальному виду

$$\begin{cases} x_1 = 2 - 0,06x_2 + 0,02x_3, \\ x_2 = 3 - 0,03x_1 + 0,05x_3, \\ x_3 = 5 - 0,01x_1 - 0,02x_2. \end{cases}$$

В матричной форме ее можно записать так:

$$\begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} = \begin{bmatrix} 2 \\ 3 \\ 5 \end{bmatrix} + \begin{bmatrix} 0 & -0,06 & 0,02 \\ -0,03 & 0 & 0,05 \\ -0,01 & 0,02 & 0 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix}.$$

Пример решения СЛУ методом итераций в системе МАТНСАД приведен ниже на рис. 24-25.

Из рисунков видно, что номер итераций, рассчитанный по теоретической формуле не всегда совпадает с практическим количеством итераций, за которые получено решение системы с заданной степенью точности \mathcal{E} .

методом

Решите систему итерационным методом

$$\text{Решение} \quad \begin{cases} 20x_1 + 3x_2 + x_3 = 68 \\ 2x_1 + 20x_2 + 3x_3 = 41 \\ 3x_1 + x_2 + 10x_3 = 60. \end{cases}$$

Разрешим уравнения относительно переменных x_1, x_2, x_3

$$\begin{aligned} 20 \cdot x_1 + 3 \cdot x_2 + x_3 &= 68 & \frac{-3}{20} \cdot x_2 - \frac{1}{20} \cdot x_3 + \frac{17}{5} \\ 2 \cdot x_1 + 20 \cdot x_2 + 3 \cdot x_3 &= 41 \\ 3 \cdot x_1 + x_2 + 10 \cdot x_3 &= 60 & \frac{-1}{10} \cdot x_1 - \frac{3}{20} \cdot x_3 + \frac{41}{20} \\ & & \frac{-3}{10} \cdot x_1 - \frac{1}{10} \cdot x_2 + 6 \end{aligned}$$

Решение ищем в виде матрицы $x = \beta + \alpha \cdot x$

Любое $(i+1)$ -е приближение вычисляется по формуле

$$\text{Образуем матрицы} \quad x^{i+1} = \beta + \alpha \cdot x^i$$

$$\alpha := \begin{pmatrix} 0 & \frac{-3}{20} & \frac{-1}{20} \\ \frac{-1}{10} & 0 & \frac{-3}{20} \\ \frac{-3}{10} & \frac{-1}{10} & 0 \end{pmatrix} \quad \beta := \begin{pmatrix} \frac{17}{5} \\ \frac{41}{20} \\ 6 \end{pmatrix}$$

Проверка достаточного условия сходимости

$$\text{norm}_1(\alpha) = 0.4 \quad \text{norm}_e(\alpha) = 0.397 \quad \text{norm}_i(\alpha) = 0.4$$

Норма матрицы меньше единицы, поэтому итерационный процесс сходится

Рис. 24

$$\hat{\alpha} = \begin{pmatrix} 2.05 \\ 6 \end{pmatrix}$$

$$i := 0..8 \quad x^{(i+1)} := \beta + \alpha \cdot x^{(i)}$$

	0	1	2	3	4	5	6	7	8	9
x =	0	3.4	2.792	3.04	2.988	3.003	2.999	3	3	3
	1	2.05	0.81	1.054	0.984	1.004	0.999	1	1	1
	2	6	4.775	5.081	4.983	5.005	4.999	5	5	5

$$\varepsilon := \frac{|x^{(7)} - x^{(6)}|}{|x^{(7)}|} \quad \varepsilon = 1.121 \times 10^{-4}$$

точность вычислений

найдем теоретически номер i нужной итерации, которая обеспечит точность 0,001 i -го приближения $x^{(i)}$

возьмем нулевое приближение и найдем его L- норму, рассчитыв ее по формуле

$$h := \begin{pmatrix} 3.4 \\ 2.05 \\ 6 \end{pmatrix} \quad |3.4| + |2.05| + |6| = 11.45$$

тогда номер i ищется по формуле

$$i \geq \frac{\lg \varepsilon + \lg(1 - |\alpha|) - \lg \|h\|}{\lg |\alpha|}$$

$$\frac{\log(\varepsilon, 10) + \log(0.6, 10) - \log(11.45, 10)}{\log(0.4, 10)} = 13.146$$

теоретически количество итераций равно 13, а практически оказалось с гарантией 6

Ответ: $x^{(6)} = \begin{pmatrix} 3 \\ 1 \\ 5 \end{pmatrix}$

Рис. 25

Вопросы для самоконтроля для самоконтроля:

1. Каковы условия сходимости итерационного процесса в приближенном методе простых итераций?
2. Что называют нормой матрицы? Какие виды норм матрицы Вы знаете?
3. Как вычислить в Mathcad нормы матрицы?
4. Каково достаточное условие сходимости метода итераций?

3.5.2 Решение систем линейных уравнений методом Зейделя

Метод Зейделя представляет собой некоторую модификацию метода итераций.

Основная его идея заключается в том, что при вычислении $(k+1)$ -го приближения неизвестной x_i , учитываются уже вычисленные ранее k -е приближения неизвестных x_1, x_2, \dots, x_{i-1} .

Пусть получена эквивалентная система (3.6). Выберем произвольно начальные приближения корней $x_1^{(0)}, x_2^{(0)}, \dots, x_n^{(0)}$. Далее, предполагая что k -ые приближения $x_n^{(k)}$ корней известны, согласно Зейделю будем строить $(k+1)$ -е приближения корней по формулам:

$$\begin{aligned}x_1^{k+1} &= \beta_1 + \alpha_{12}x_2^k + \alpha_{13}x_3^k + \dots + \alpha_{1n}x_n^k \\x_2^{k+1} &= \beta_2 + \alpha_{21}x_1^k + \alpha_{23}x_3^k + \dots + \alpha_{2n}x_n^k \\x_n^{k+1} &= \beta_n + \alpha_n x_1^k + \alpha_n x_2^k + \dots + \alpha_{nn-1}x_{n-1}^k\end{aligned}$$

Заметим, что указанные выше условия сходимости для простой итерации остается верной для итерации по методу Зейделя. Обычно метод Зейделя дает *лучшую сходимость*, чем метод простой итерации, но приводит к более громоздким вычислениям.

Пример решения методом Зейделя системы уравнений с применением MathCAD представлен на рис. 26-27.

Решение системы линейных уравнений методом Зейделя

$$A := \begin{pmatrix} 100 & 6 & -2 \\ 6 & 200 & -10 \\ 1 & 2 & 100 \end{pmatrix} \quad B := \begin{pmatrix} 200 \\ 600 \\ 500 \end{pmatrix}$$

$$100 \cdot x_1 + 6x_2 + -2x_3 = 200 \quad \frac{-3}{50} \cdot x_2 + \frac{1}{50} \cdot x_3 + 2$$

$$6 \cdot x_1 + 200 \cdot x_2 - 10x_3 = 600 \quad \frac{-3}{100} \cdot x_1 + \frac{1}{20} \cdot x_3 + 3$$

$$x_1 + 2 \cdot x_2 + 100 \cdot x_3 = 500$$

$$\frac{-1}{100} \cdot x_1 - \frac{1}{50} \cdot x_2 + 5$$

$$\alpha := \begin{pmatrix} 0 & \frac{-3}{50} & \frac{1}{50} \\ \frac{-3}{100} & 0 & \frac{1}{20} \\ \frac{-1}{100} & \frac{1}{-50} & 0 \end{pmatrix}$$

$$\alpha = \begin{pmatrix} 0 & -0.06 & 0.02 \\ -0.03 & 0 & 0.05 \\ -0.01 & -0.02 & 0 \end{pmatrix}$$

$$\beta := \begin{pmatrix} 2 \\ 3 \\ 5 \end{pmatrix}$$

$$\text{norm}_1(\alpha) = 0.08 \quad \text{norm}_e(\alpha) = 0.089 \quad \text{norm}_i(\alpha) = 0.08$$

$$i := 0..8 \quad x^{(0)} := \beta$$

$$x^{(0)} = \begin{pmatrix} 2 \\ 3 \\ 5 \end{pmatrix}$$

$$x^{(i+1)} := \beta + \alpha \cdot x^{(i)}$$

Итерационная формула Зейделя

$$A_1 := \begin{pmatrix} 0 & 0 & 0 \\ -0.03 & 0 & 0 \\ -0.01 & -0.02 & 0 \end{pmatrix} \quad A_2 := \begin{pmatrix} 0 & -0.06 & 0.02 \\ 0 & 0 & 0.05 \\ 0 & 0 & 0 \end{pmatrix}$$

Создание дополнительных матриц

Рис. 26


```

zeidel(a, c, ε) :=
  n ← rows(a)
  it ← 0
  max_it ← 100
  for i ∈ 1..n
    for j ∈ 1..n
      bi,j ←  $\frac{-a_{i,j}}{a_{i,i}}$  if i ≠ j
      bi,j ← 0 otherwise
    gi ←  $\frac{c_i}{a_{i,i}}$ 
    xi ← gi
  while 1
    y ← x
    it ← it + 1
    for i ∈ 1..n
      s ← 0
      for j ∈ 1..n
        s ← s + bi,j · xj
      xi ← s + gi
    break if |y - x| < ε ∨ it ≥ max_it
  error("Нет сходимости!") if it ≥ max_it
  x

```

Рис. 28

Обратите внимание на условие окончания итераций $|y - x| < \varepsilon$. Здесь запись $|y - x|$ означает вычисление евклидовой нормы вектора, т.е. расстояния между точками в n -мерном

пространстве. Вычисления проводятся по формуле Евклидовой формы матрицы $|y - x| = \sqrt{\sum_{i=1}^n (y_i - x_i)^2}$.

Применим функцию `zeidel` для поиска приближенного решения системы, матрица которой является матрицей с диагональным преобладанием, что обеспечивает сходимость метода Зейделя.

$$\begin{aligned} 5x_1 + x_2 + 0.5x_3 &= 3 \\ -x_1 + 8x_2 + 2x_3 &= 0.5 \\ 0.1x_1 - x_2 + 3x_3 &= 2 \end{aligned}$$

$$\underline{\underline{A}} := \begin{pmatrix} 5 & 1 & 0.5 \\ -1 & 8 & 2 \\ 0.1 & -1 & 3 \end{pmatrix} \quad \underline{\underline{c}} := \begin{pmatrix} 3 \\ 0.5 \\ 2 \end{pmatrix}$$

$$x := \text{zeidel}(A, c, 0.001)$$

$$x = \begin{pmatrix} 0.542 \\ -0.029 \\ 0.639 \end{pmatrix} \quad \text{Проверка: } A \cdot x = \begin{pmatrix} 3 \\ 0.5 \\ 2 \end{pmatrix}$$

Лабораторная работа № 3

Решение систем линейных методом простых итераций и методом Зейделя

Цель работы: изучить методы итераций, метод Зейделя.

Задание 1. Решите систему линейных уравнений методом итераций.

Задание 2. Решите систему линейных уравнений методом Зейделя.

Оцените погрешность вычислений.

На каком шаге итераций погрешность вычислений минимальна?

Вариант	Задание 1	Задание 2
1	$\begin{cases} 24.3x_1 - 2.9x_2 - 1.3x_3 + 5.2x_4 = 8.3, \\ 5.6x_1 - 38.3x_2 + 9.2x_3 - 3.5x_4 = 4.5, \\ 6.3x_1 + 8.6x_2 - 22.1x_3 + 8.9x_4 = 8.1, \\ 7.9x_1 - 7.5x_2 + 4.5x_3 + 27.2x_4 = 5.7. \end{cases}$	$\begin{cases} 100.51x - 1.9y - 0.3z = 1.1, \\ 1.9x + 103.2y - 0.5z = 1.2, \\ 3.6x + 100.3y - 1.4z = 5.2. \end{cases}$
2	$\begin{cases} 10.7x_1 - 0.3x_2 + 0.8x_3 + 0.1x_4 = 0.4, \\ 0.9x_1 + 11.4x_2 - 0.4x_3 - 0.4x_4 = 0.2, \\ 0.8x_1 + 1.6x_2 - 10.8x_3 + 0.7x_4 = 0.9, \\ 0.2x_1 - 0.4x_2 - 0.2x_3 - 10.4x_4 = 0.5. \end{cases}$	$\begin{cases} 10.51x - 0.29y + 0.13z = 1.4 \\ 0.95x - 10.32y + 1.43z = 4.9 \\ 0.71x - 0.94y + 0.026z = 1.7 \end{cases}$
3	$\begin{cases} 141.1x_1 - 3.2x_2 + 5.1x_3 + 5.8x_4 = 3.9, \\ 6.6x_1 - 181.4x_2 + 8.2x_3 - 9.9x_4 = 7.2, \\ 9.3x_1 - 7.6x_2 + 191.1x_3 - 3.8x_4 = 8.1, \\ 7.9x_1 - 5.5x_2 + 4.2x_3 - 141.7x_4 = 9.1. \end{cases}$	$\begin{cases} 10.24x + 5.29y - 7.21z = 5.9 \\ 0.96x - 10.42y + 3.69z = 8.6 \\ 3.71x - 7.47y - 10.22z = 9.4 \end{cases}$
4	$\begin{cases} 12.6x_1 - 3.1x_2 - 2.7x_3 - 2.5x_4 = 1.6, \\ 4.6x_1 - 11.4x_2 - 1.2x_3 - 3.7x_4 = -4.8, \\ 1.3x_1 - 6.6x_2 - 14.1x_3 - 1.6x_4 = -2.2, \\ 3.9x_1 + 2.7x_2 - 2.8x_3 - 11.3x_4 = 0.3. \end{cases}$	$\begin{cases} -100.3x - 0.9y + 0.3z = 1.61 \\ -0.5x + 100.2y - 0.8z = 1.59 \\ -0.1x - 0.7y + 100.2z = 2.7 \end{cases}$
5	$\begin{cases} 17.8x_1 + 9.5x_2 - 3.7x_3 - 9.3x_4 = 7.4, \\ 4.9x_1 - 14.8x_2 - 6.9x_3 - 7.6x_4 = -4.9, \\ 7.6x_1 - 3.6x_2 - 19.1x_3 + 6.6x_4 = 6.2, \\ 8.4x_1 + 4.7x_2 - 2.8x_3 - 11.3x_4 = 4.3. \end{cases}$	$\begin{cases} 10.8x + 5.6y - 2.6z = 8.12, \\ 8.5x - 10.9y + 5.8z = 9.5, \\ 9.4x + 0.6y - 10.7z = 5.13. \end{cases}$
6	$\begin{cases} 10.6x_1 - 0.1x_2 - 0.7x_3 - 0.5x_4 = 1.8, \\ 4.9x_1 - 17.4x_2 - 8.2x_3 + 9.2x_4 = 3.9, \\ 1.4x_1 + 1.6x_2 + 11.1x_3 - 1.3x_4 = 5.9, \\ 8.2x_1 + 9.7x_2 - 6.8x_3 + 17.9x_4 = 4.6. \end{cases}$	$\begin{cases} 10.3x - 6.3y - 6.3z = 8.3, \\ 8.5x + 10.2y - 9.8z = 4.5, \\ 7.3x + 3.7y + 10.8z = 9.1. \end{cases}$

$$7 \quad \begin{cases} 12.6x_1 - 5.1x_2 + 0.7x_3 - 0.5x_4 = 2.8, \\ 4.9x_1 - 17.4x_2 - 8.2x_3 + 9.2x_4 = 3.9, \\ 5.4x_1 + 1.6x_2 + 11.1x_3 - 1.3x_4 = 3.9, \\ 8.2x_1 + 6.7x_2 - 6.8x_3 + 17.9x_4 = 2.6. \end{cases} \quad \begin{cases} 10.6x - 0.6y + 0.9z = 1.6, \\ 1.9x - 10.9y - 0.4z = 1.3, \\ 2.3x + 0.6y + 10.7z = 2.2. \end{cases}$$

$$8 \quad \begin{cases} 12.6x_1 - 3.1x_2 - 2.7x_3 - 2.5x_4 = 1.6, \\ 4.6x_1 - 11.4x_2 - 1.2x_3 - 3.7x_4 = -4.8, \\ 1.3x_1 - 6.6x_2 - 14.1x_3 - 1.6x_4 = -2.2, \\ 3.9x_1 + 2.7x_2 - 2.8x_3 - 11.3x_4 = 0.3. \end{cases} \quad \begin{cases} 10.6x - 7.9y - 8.3z = 2.11, \\ -3.5x + 10.2y - 3.8z = 3.57, \\ 5.1x + 7.7y - 10.2z = 6.93. \end{cases}$$

$$9 \quad \begin{cases} 10.7x_1 - 0.3x_2 + 0.8x_3 + 0.1x_4 = 0.4, \\ 0.9x_1 + 11.4x_2 - 0.4x_3 - 0.4x_4 = 0.2, \\ 0.8x_1 + 1.6x_2 - 12.8x_3 + 0.7x_4 = 0.9, \\ 0.2x_1 - 0.4x_2 - 0.2x_3 - 10.4x_4 = 0.5. \end{cases} \quad \begin{cases} 10.51x - 0.2y + 0.13z = 1.4, \\ 0.95x - 10.3y + 1.43z = 4.9, \\ 0.71x - 0.94y + 10.26z = 1.7. \end{cases}$$

$$10 \quad \begin{cases} 14.1x_1 - 3.2x_2 + 5.1x_3 + 5.8x_4 = 3.9, \\ 6.6x_1 - 18.4x_2 + 8.2x_3 - 9.9x_4 = 7.2, \\ 9.3x_1 - 7.6x_2 + 19.1x_3 - 3.8x_4 = 8.1, \\ 7.9x_1 - 5.5x_2 + 4.2x_3 - 14.7x_4 = 9.1. \end{cases} \quad \begin{cases} 10.24x + 5.29y - 7.2z = 5.9, \\ 0.96x - 10.4y + 3.6z = 8.6, \\ 3.71x - 7.47y - 10.2z = 9.4. \end{cases}$$

$$11 \quad \begin{cases} 15.1x_1 - 3.2x_2 + 5.1x_3 + 5.8x_4 = 13.9, \\ 10.6x_1 - 18.4x_2 + 8.2x_3 - 9.9x_4 = 7.2, \\ 9.3x_1 - 7.6x_2 + 19.1x_3 - 3.8x_4 = 18.1, \\ 5.9x_1 - 5.5x_2 + 4.2x_3 - 14.7x_4 = 9.1. \end{cases} \quad \begin{cases} 10.24x + 5.29y - 7.1z = 5.9, \\ 2.96x + 11.42y + 3.9z = 1.6, \\ 4.71x - 7.47y - 15.2z = 0.4. \end{cases}$$

$$12 \quad \begin{cases} 11.8x_1 + 9.5x_2 - 3.7x_3 - 9.3x_4 = 7.4, \\ 0.9x_1 - 14.8x_2 - 6.9x_3 - 7.6x_4 = -4.9, \\ 7.6x_1 - 3.6x_2 - 19.1x_3 + 6.6x_4 = 1.2, \\ 8.4x_1 + 4.7x_2 - 2.8x_3 - 11.3x_4 = 4.3. \end{cases} \quad \begin{cases} 109.24x + 5.29y - 7.21z = 5.9, \\ 0.96x - 10.442y + 3.69z = 8.6, \\ 3.71x - 7.47y - 10.522z = 9.4. \end{cases}$$

$$13 \quad \begin{cases} 14.1x_1 - 3.2x_2 + 5.1x_3 + 5.8x_4 = 3.9, \\ 6.6x_1 - 18.4x_2 + 8.2x_3 - 9.9x_4 = 7.2, \\ 9.3x_1 - 7.6x_2 + 19.1x_3 - 3.8x_4 = 2.1, \\ 7.6x_1 + 5.5x_2 + 4.2x_3 - 14.7x_4 = 9.1. \end{cases} \quad \begin{cases} 106.2x + 5.29y - 7.21z = 5.9 \\ 11.96x - 104.42y + 3.69z = 0. \\ 3.71x - 7.47y - 105.22z = 9. \end{cases}$$

$$14 \quad \begin{cases} 18.7x_1 - 0.3x_2 + 0.8x_3 + 0.1x_4 = 0.4, \\ 0.9x_1 + 11.4x_2 - 0.4x_3 - 0.4x_4 = 1.2, \\ 7.8x_1 + 1.6x_2 - 10.8x_3 + 0.7x_4 = 0.9, \\ 6.2x_1 - 0.4x_2 - 0.2x_3 - 10.4x_4 = 0.5. \end{cases} \quad \begin{cases} 107.8x + 5.6y - 2.6z = 8.12, \\ 8.5x + 106.9y + 5.8z = 2.5, \\ 5.4x + 0.6y - 102.7z = 5.13. \end{cases}$$

$$15 \quad \begin{cases} 10.7x_1 - 0.3x_2 + 0.8x_3 + 0.1x_4 = 0.4, \\ 9.9x_1 + 10.4x_2 - 0.4x_3 - 0.4x_4 = 1.2, \\ 7.8x_1 + 1.6x_2 - 10.8x_3 + 0.7x_4 = 0.9, \\ 6.2x_1 - 0.7x_2 + 0.2x_3 - 10.4x_4 = 1.5. \end{cases} \quad \begin{cases} 107.8x + 5.6y - 2.6z = 8.12, \\ 8.5x + 106.9y + 5.8z = 2.5, \\ 5.4x + 0.6y - 102.7z = 5.13. \end{cases}$$

ГЛАВА 4

ИНТЕРПОЛИРОВАНИЕ ФУНКЦИЙ

4.1 Интерполяция

Пусть на некотором отрезке заданы $n + 1$ точки $x_0, x_1, x_2, \dots, x_n$, которые называются *узлами* интерполяции, и значения некоторой функции $y = f(x)$ в этих точках: $y_0 = f(x_0), y_1 = f(x_1), \dots, y_n = f(x_n)$. Тогда функцию $y = f(x)$ можно определить таблицей

x_i	x_0	x_1	...	x_n
y_i	y_0	y_1	...	y_n

Требуется построить функцию $F(x)$, принадлежащую известному классу функций и принимающую в узлах интерполяции те же значения, что и $f(x)$, т.е. такую, что

$$y_0 = F(x_0), \quad y_1 = F(x_1), \quad \dots, \quad y_n = F(x_n).$$

Такая функция $F(x)$ называется *интерполирующей функцией*.

Геометрически это означает, что нужно найти кривую $y = F(x)$ некоторого определённого типа, проходящую через заданные точки $N_i(x_i, y_i)$, где $i = 0, 1, \dots, n$.

Заметим, что в такой общей постановке задача интерполирования может иметь бесконечное множество решений или совсем не иметь решений. Однако задача становится однозначной, если вместо произвольной функции $F(x)$ следует искать многочлен $P_n(x)$ степени n такой, что

$$y_0 = P_n(x_0), \quad y_1 = P_n(x_1), \quad \dots, \quad y_n = P_n(x_n). \quad (4.1)$$

Таким образом, *задачей интерполяции* является построение многочлена $P_n(x)$, значения которого в узлах интерполяции x_i равны соответствующим значениям заданной функции, т.е. для него выполняются условия (4.1).

На практике полученную интерполяционную формулу $y = P_n(x)$ используют для приближённого вычисления значений заданной функции

$f(x)$, где значения аргумента x отличны от узлов интерполяции. Такая операция называется *интерполированием* функции $f(x)$.

4.1.1 Интерполяционный многочлен Лагранжа

Интерполяционным многочленом Лагранжа называется многочлен

$$P_n(x) = \sum_{k=0}^n y_k \frac{(x-x_0)(x-x_1)\dots(x-x_{k-1})(x-x_{k+1})\dots(x-x_n)}{(x_k-x_0)(x_k-x_1)\dots(x_k-x_{k-1})(x_k-x_{k+1})\dots(x_k-x_n)}. \quad (4.2)$$

Этот многочлен удовлетворяет условиям (2.1).

Интерполяционной формулой Лагранжа называется формула

$$f(x) \approx \sum_{k=0}^n y_k \frac{(x-x_0)(x-x_1)\dots(x-x_{k-1})(x-x_{k+1})\dots(x-x_n)}{(x_k-x_0)(x_k-x_1)\dots(x_k-x_{k-1})(x_k-x_{k+1})\dots(x_k-x_n)}. \quad (4.3)$$

Если функция $f(x)$ на отрезке $[x_0, x_n]$ определена и имеет непрерывные производные до $(n+1)$ -го порядка включительно, то погрешность интерполяционной формулы в каждой точке этого отрезка оценивается неравенством

$$|f(x) - P_n(x)| \leq \frac{M_{n+1}}{(n+1)!} |R_{n+1}(x)|, \quad (4.4)$$

где $M_{n+1} = \max_{x_0 \leq x \leq x_n} |f^{(n+1)}(x)|$, $R_{n+1}(x) = (x-x_0)(x-x_1)\dots(x-x_n)$.

Пример 1.

Найти интерполяционный многочлен Лагранжа $P(x)$, для которого $P(-1) = 12$, $P(1) = -3$, $P(3) = 4$.

Решение. В данном случае $x_0 = -1$, $x_1 = 1$, $x_2 = 3$, $y_0 = 12$, $y_1 = -3$, $y_2 = 4$. При $n = 2$ формула (2.2) принимает вид

$$P_2(x) = y_0 \frac{(x-x_1)(x-x_2)}{(x_0-x_1)(x_0-x_2)} + y_1 \frac{(x-x_0)(x-x_2)}{(x_1-x_0)(x_1-x_2)} + y_2 \frac{(x-x_0)(x-x_1)}{(x_2-x_0)(x_2-x_1)}.$$

Подставляя в эту формулу заданные значения, находим

$$\begin{aligned} P_2(x) &= 12 \frac{(x-1)(x-3)}{(-1-1)(-1-3)} - 3 \frac{(x+1)(x-3)}{(1+1)(1-3)} + 4 \frac{(x+1)(x-1)}{(3+1)(3-1)} = \frac{12}{8}(x^2 - \\ &- 4x + 3) + \frac{3}{4}(x^2 - 2x - 3) + \frac{4}{8}(x^2 - 1) = 4(6x^2 - 24x + 18 + 3x^2 - 6x - 9 + \\ &+ 2x^2 - 2) = 4(11x^2 - 30x + 7) = 44x^2 - 120x + 28. \end{aligned}$$

Пример 2.

Найти интерполяционный многочлен Лагранжа $P(x)$ для функции, заданной таблицей

x	0	1	2	3
y	6	0	-10	-30

Решение. В данном случае $x_0 = 0$, $x_1 = 1$, $x_2 = 2$, $x_3 = 3$, $y_0 = 6$, $y_1 = 0$, $y_2 = -10$, $y_3 = -30$.

При $n = 3$ формула (2.2) принимает вид

$$P_3(x) = y_0 \frac{(x-x_1)(x-x_2)(x-x_3)}{(x_0-x_1)(x_0-x_2)(x_0-x_3)} + y_1 \frac{(x-x_0)(x-x_2)(x-x_3)}{(x_1-x_0)(x_1-x_2)(x_1-x_3)} +$$

$$+ y_2 \frac{(x-x_0)(x-x_1)(x-x_3)}{(x_2-x_0)(x_2-x_1)(x_2-x_3)} + y_3 \frac{(x-x_0)(x-x_1)(x-x_2)}{(x_3-x_0)(x_3-x_1)(x_3-x_2)}$$

Подставляя в формулу заданные значения, находим

$$P_3(x) = 6 \frac{(x-1)(x-2)(x-3)}{(0-1)(0-2)(0-3)} - 10 \frac{(x-0)(x-1)(x-3)}{(2-0)(2-1)(2-3)} - 30 \frac{(x-0)(x-1)(x-2)}{(3-0)(3-1)(3-2)} =$$

$$= (x-1)(-x^2 + 5x - 6) + 5(x-1)(x^2 - 3x) - 5(x-1)(x^2 - 2x) = (x-1)(-x^2 - 6) = -x^3 - 6x + x^2 + 6 = -x^3 + x^2 - 6x + 6.$$

Пример 3.

Найти интерполяционный многочлен Лагранжа 2-го порядка для функции $y = \cos x$ на отрезке $[0, \pi/4]$, если заданы значения функции в трёх узлах интерполяции:

x	0	$\pi/6$	$\pi/4$
$y = \cos x$	1	$\frac{\sqrt{3}}{2}$	$\frac{\sqrt{2}}{2}$

С помощью интерполяционной формулы вычислить приближённое значение $\cos \frac{\pi}{12}$ и оценить погрешность результата вычислений.

Решение. В нашем случае $x_0 = 0$, $x_1 = \pi/6$, $x_2 = \pi/4$,

$$y_0 = 1, \quad y_1 = \frac{\sqrt{3}}{2}, \quad y_2 = \frac{\sqrt{2}}{2}.$$

Многочлен Лагранжа для трёх узлов интерполяции запишется так:

$$P_2(x) = y_0 \frac{(x-x_1)(x-x_2)}{(x_0-x_1)(x_0-x_2)} + y_1 \frac{(x-x_0)(x-x_2)}{(x_1-x_0)(x_1-x_2)} + y_2 \frac{(x-x_0)(x-x_1)}{(x_2-x_0)(x_2-x_1)}.$$

Подставляя в формулу заданные значения, находим

$$\begin{aligned}
P_2(x) &= 1 \frac{(x - \frac{\pi}{6})(x - \frac{\pi}{4})}{(-\frac{\pi}{6})(-\frac{\pi}{4})} + \frac{\sqrt{3}}{2} \frac{(x-0)(x - \frac{\pi}{4})}{(\frac{\pi}{6} - 0)(\frac{\pi}{6} - \frac{\pi}{4})} + \frac{\sqrt{2}}{2} \frac{(x-0)(x - \frac{\pi}{6})}{(\frac{\pi}{4} - 0)(\frac{\pi}{4} - \frac{\pi}{6})} = \\
&= \frac{24}{\pi^2} (x^2 - \frac{5}{12} \pi x + \frac{\pi^2}{24}) - \frac{\sqrt{3}}{2} \cdot \frac{72}{\pi^2} (x^2 - \frac{\pi}{4} x) + \frac{\sqrt{2}}{2} \cdot \frac{48}{\pi^2} (x^2 - \frac{\pi}{6} x) = \\
&= \frac{x^2}{\pi^2} (24 - 36\sqrt{3} + 24\sqrt{2}) + \frac{x}{\pi} (-10 + 9\sqrt{3} - 4\sqrt{2}) + 1 \approx -0.4471x^2 - 0.022x + 1.
\end{aligned}$$

При $x = \frac{\pi}{12} \approx 0.2618$ $P_2(\frac{\pi}{12}) \approx 0.9636$.

С помощью неравенства (4.4) оценим погрешность вычислений:

$$\left| \cos \frac{\pi}{12} - P_2\left(\frac{\pi}{12}\right) \right| \leq \frac{M_3}{3!} \cdot \left| R_3\left(\frac{\pi}{12}\right) \right|,$$

где $R_3(x) = (x - x_0)(x - x_1)(x - x_2) = x(x - \pi/6)(x - \pi/4)$,
 $R_3(\pi/12) = (\pi/12)(-\pi/12)(-\pi/6) = 2(\pi/12)^3$.

Так как $y = \cos x$, $y' = -\sin x$, $y'' = -\cos x$, $y''' = \sin x$,
то

$$M_3 = \max_{0 \leq x \leq \pi/4} |f'''(x)| = \max_{0 \leq x \leq \pi/4} |\sin x| = \frac{\sqrt{2}}{2} \approx 0.707,$$

следовательно,

$$\left| \cos \frac{\pi}{12} - P_2\left(\frac{\pi}{12}\right) \right| \leq \frac{\sqrt{2} \cdot 2}{2 \cdot 3!} \left(\frac{\pi}{12}\right)^3 = \frac{\sqrt{2} \cdot \pi^3}{6 \cdot 12^3} \approx 4.229 \cdot 10^{-3}.$$

Таким образом, $\cos \frac{\pi}{12} \approx 0.9636 \pm 0.0042$. Заметим, что это

значение с шестью верными цифрами: $\cos \frac{\pi}{12} = 0.965926$.

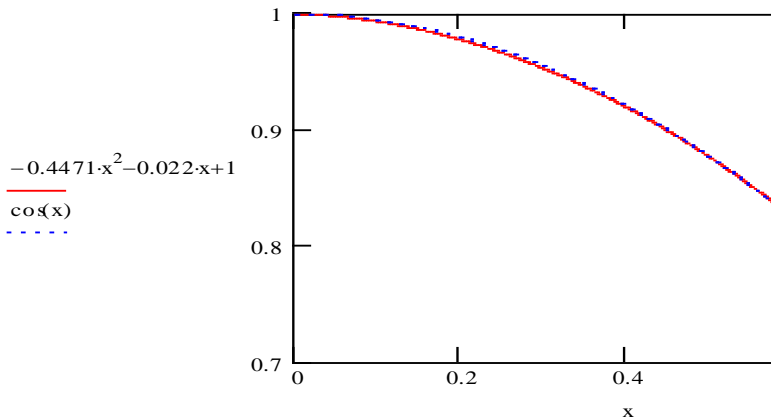


Рис. 29. Графики функций $y = \cos x$ и $y = P_2(x)$

На рис. 29 представлены графики функций $y = P_2(x)$ и $y = \cos x$ на отрезке $[0, \pi/4]$, выполненные в системе MathCAD.

Конечные разности

Рассмотрим значения функции $y = f(x)$ в точках $x_0, x_1, x_2, \dots, x_n$:

$$y_0 = f(x_0), \quad y_1 = f(x_1), \quad \dots, \quad y_n = f(x_n).$$

Выделим всевозможные пары соседних значений функции, в каждом случае вычтем предыдущее значение из последующего, получим разности:

$$y_1 - y_0, \quad y_2 - y_1, \quad \dots, \quad y_{n-1} - y_n.$$

Они называются *конечными разностями первого порядка* или просто *первыми разностями* и обозначаются:

$$\Delta y_0 = y_1 - y_0, \quad \Delta y_1 = y_2 - y_1, \quad \dots, \quad \Delta y_{n-1} = y_n - y_{n-1} \quad (4.5)$$

или кратко: $\Delta y_i = y_{i+1} - y_i$ ($i = 0, 1, \dots, n-1$).

Разностями второго порядка или вторыми разностями называют разности первых разностей и обозначают:

$$\begin{aligned}\Delta^2 y_0 &= \Delta y_1 - \Delta y_0, \\ \Delta^2 y_1 &= \Delta y_2 - \Delta y_1,\end{aligned}$$

⋮
(4.6)

$$\Delta^2 y_{n-1} = \Delta y_n - \Delta y_{n-1}$$

или кратко: $\Delta^2 y_i = \Delta y_{i+1} - \Delta y_i \quad (i = 0, 1, \dots, n-1)$.

Разностями третьего порядка или третьими разностями называют разности вторых разностей и обозначают:

$$\begin{aligned}\Delta^3 y_0 &= \Delta^2 y_1 - \Delta^2 y_0, \\ \Delta^3 y_1 &= \Delta^2 y_2 - \Delta^2 y_1,\end{aligned}$$

(4.7)

$$\Delta^3 y_{n-1} = \Delta^2 y_n - \Delta^2 y_{n-1}$$

или кратко: $\Delta^3 y_i = \Delta^2 y_{i+1} - \Delta^2 y_i \quad (i = 0, 1, \dots, n-1)$.

Аналогично определяются последующие разности.

Конечные разности любого порядка могут быть представлены через значения функции. Для разностей первого порядка это следует из определения.

Для разностей второго порядка имеем

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

Аналогично для разностей третьего порядка

$$\begin{aligned}\Delta^3 y_i &= \Delta^2 y_{i+1} - \Delta^2 y_i = (y_{i+3} - 2y_{i+2} + y_{i+1}) - (y_{i+2} - 2y_{i+1} + y_i) = \\ &= y_{i+3} - 3y_{i+2} + 3y_{i+1} - y_i \text{ и т.д.}\end{aligned}$$

Таблица разностей различных порядков строится согласно схемы (рис. 30). Каждое число этой таблицы (начиная с третьего столбца) является разностью двух смежных чисел столбца слева (из нижнего числа вычитается верхнее; разность записывается в

следующем столбце между этими числами). Третий столбец содержит первые разности, четвёртый - вторые и т.д.

x	y	Δy	$\Delta^2 y$	$\Delta^3 y$	$\Delta^4 y$	$\Delta^5 y$
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	$\Delta^2 y_1$	$\Delta^3 y_1$	$\Delta^4 y_0$	
x_3	y_3	Δy_3	$\Delta^2 y_2$	$\Delta^3 y_2$	$\Delta^4 y_1$	$\Delta^5 y_0$
x_4	y_4	Δy_4	$\Delta^2 y_3$
x_5	y_5			
...	...					

Рис. 30. Схема построения конечных разностей

Для контроля вычислений при составлении таблицы разностей пользуются следующим утверждением:

сумма чисел в каждом столбце разностей равна разности крайних чисел предыдущего столбца. Например,

$$\Delta y_0 + \Delta y_1 + \Delta y_2 + \dots + \Delta y_n = (y_1 - y_0) + (y_2 - y_1) + \dots + (y_{n+1} - y_n) = y_{n+1} - y_0.$$

Обычно все разности в таблице записывают целыми числами или в единицах младшего разряда значений функции.

Заметим, что конечные разности n -го порядка от многочлена степени n постоянны, а конечные разности $(n + 1)$ -го порядка равны нулю. Это свойство даёт простой способ составления таблиц. Непосредственно вычисляем значения многочлена для $n + 1$ значений аргумента и составляем таблицу, в которую входят раз-

ности до n -го порядка включительно. Далее, пользуясь тем, что разности n -го порядка постоянны, продолжаем столбец разностей $(n - 1)$ -го порядка. Для получения новых чисел этого столбца складываем соответствующие разности $(n - 1)$ -го порядка с разностями n -го порядка. Затем последовательно продолжаем столбцы разностей $(n - 2)$ -го, $(n - 3)$ -го порядков и т.д., пока не получим продолжение столбца $y_i = f(x_i)$, т.е. значений многочлена.

Пример 4.

Составить таблицу конечных разностей для следующих значений x и $y = f(x)$: $x_0 = -2$, $x_1 = -1$, $x_2 = 0$, $x_3 = 1$, $x_4 = 2$,

$$y_0 = 4, \quad y_1 = 1, \quad y_2 = 0, \quad y_3 = 1, \quad y_4 = 4.$$

Решение. По формулам (2.5) найдём первые разности:

$$\Delta y_0 = y_1 - y_0 = 1 - 4 = -3;$$

$$\Delta y_1 = y_2 - y_1 = 0 - 1 = -1;$$

$$\Delta y_2 = y_3 - y_2 = 1 - 0 = 1;$$

$$\Delta y_3 = y_4 - y_3 = 4 - 1 = 3.$$

По формулам (2.6) получим вторые разности:

$$\Delta^2 y_0 = \Delta y_1 - \Delta y_0 = -1 + 3 = 2;$$

$$\Delta^2 y_1 = \Delta y_2 - \Delta y_1 = 1 + 1 = 2;$$

$$\Delta^2 y_2 = \Delta y_3 - \Delta y_2 = 3 - 1 = 2.$$

По формулам (2.7) найдём третьи разности:

$$\Delta^3 y_0 = \Delta^2 y_1 - \Delta^2 y_0 = 2 - 2 = 0; \quad \Delta^3 y_1 = \Delta^2 y_2 - \Delta^2 y_1 = 2 - 2 = 0.$$

Конечная разность четвёртого порядка

$$\Delta^4 y_0 = \Delta^3 y_1 - \Delta^3 y_0 = 0 - 0 = 0.$$

Полученные значения занесём в таблицу.

x	y	Δy	$\Delta^2 y$	$\Delta^3 y$	$\Delta^4 y$
-2	4				
		-3			
-1	1		2		
		-1		0	
0	0		2		0
		1		0	
1	1		2		
		3			
2	4				
Σ		0	6	0	0
S	0	6	0		

Две последние строки записаны для контроля вычислений: в строке Σ числа равны суммам чисел, стоящих в соответствующем столбце, в строке **S** - разности последнего и первого чисел соответствующего столбца. Совпадение этих чисел по диагонали означает, что вычисления проведены верно.

Разделённые разности

Разделённые разности первого порядка определяются формулами:

$$f(x_1, x_0) = \frac{y_1 - y_0}{x_1 - x_0}; f(x_2, x_1) = \frac{y_2 - y_1}{x_2 - x_1}; \dots, f(x_n, x_{n-1}) = \frac{y_n - y_{n-1}}{x_n - x_{n-1}} \quad (4.8)$$

Разделённые разности второго порядка получаются из разделённых разностей первого порядка по формулам:

$$f(x_2, x_1, x_0) = \frac{f(x_2, x_1) - f(x_1, x_0)}{x_2 - x_0}; \quad f(x_3, x_2, x_1) = \frac{f(x_3, x_2) - f(x_2, x_1)}{x_3 - x_1} \quad (4.9)$$

Аналогично определяются *разделённые разности третьего порядка*:

$$f(x_3, x_2, x_1, x_0) = \frac{f(x_3, x_2, x_1) - f(x_2, x_1, x_0)}{x_3 - x_0};$$

$$f(x_4, x_3, x_2, x_1) = \frac{f(x_4, x_3, x_2) - f(x_3, x_2, x_1)}{x_4 - x_1}. \quad (4.10)$$

Разделённые разности n -го порядка получаются из разделённых разностей $(n - 1)$ -го порядка по формуле

$$f(x_n, x_{n-1}, \dots, x_1, x_0) = \frac{f(x_n, x_{n-1}, \dots, x_1) - f(x_{n-1}, x_{n-2}, \dots, x_0)}{x_n - x_0}.$$

В случае равноотстоящих узлов с шагом h разделённые разности различных порядков имеют вид:

$$f(x_1, x_0) = \frac{\Delta y_0}{h}; \quad f(x_2, x_1) = \frac{\Delta y_1}{h}; \quad \dots, \quad f(x_n, x_{n-1}) = \frac{\Delta y_{n-1}}{h}; \quad (4.11)$$

$$f(x_2, x_1, x_0) = \frac{\Delta^2 y_0}{2!h^2}, \quad f(x_3, x_2, x_1) = \frac{\Delta^2 y_1}{2!h^2}; \quad f(x_n, x_{n-1}, \dots, x_1, x_0) = \frac{\Delta^n y_0}{n!h^n}.$$

Пример 5.

Составить таблицу разделённых разностей для следующих значений x и $y = f(x)$:

$$x_0 = -3, \quad x_1 = -2, \quad x_2 = -1, \quad x_3 = 1, \quad x_4 = 2,$$

$$y_0 = -9, \quad y_1 = -16, \quad y_2 = -3, \quad y_3 = 11, \quad y_4 = 36.$$

Решение. По формулам (4.8) найдём разделённые разности первого порядка:

$$f(x_1, x_0) = -7, \quad f(x_2, x_1) = 13, \quad f(x_3, x_2) = 7, \quad f(x_4, x_3) = 25.$$

По формулам (4.9) получим разделённые разности второго порядка:

$$f(x_2, x_1, x_0) = 10, \quad f(x_3, x_2, x_1) = -2, \quad f(x_4, x_3, x_2) = 6.$$

По формулам (4.10) получим разделённые разности третьего порядка:

$$f(x_3, x_2, x_1, x_0) = -3, \quad f(x_4, x_3, x_2, x_1) = 2.$$

Разделённая разность четвёртого порядка

$$f(x_4, x_3, x_2, x_1, x_0) = \frac{f(x_4, x_3, x_2, x_1) - f(x_3, x_2, x_1, x_0)}{x_4 - x_1} = \frac{2 - (-3)}{2 - (-3)} = 1.$$

Полученные значения занесём в таблицу.

x	y	Разделённые разности			
-3	-9				
		-7			
-2	-16		10		
		13		-3	
-1	-3		-2		1
		7		2	
1	11		6		
		25			
2	36				

4.1.2 Интерполяционный многочлен Ньютона

Определение Интерполяционным многочленом Ньютона называется многочлен

$$P_n(x) = y_0 + (x - x_0)f(x_0, x_1) + (x - x_0)(x - x_1)f(x_0, x_1, x_2) + \dots \\ \dots + (x - x_0)(x - x_1) \dots (x - x_{n-1})f(x_0, x_1, \dots, x_n), \quad (4.12)$$

в котором $f(x_0, x_1), f(x_0, x_1, x_2), \dots, f(x_0, x_1, \dots, x_n)$ - разделённые разности различных порядков. Этот многочлен удовлетворяет условиям (4.1).

В силу единственности интерполяционного многочлена n -й степени интерполяционный многочлен Ньютона перегруппировкой членов можно преобразовать в интерполяционный многочлен Лагранжа и обратно.

Определение Интерполяционной формулой Ньютона называется формула

$$f(x) \approx y_0 + (x - x_0)f(x_0, x_1) + (x - x_0)(x - x_1)f(x_0, x_1, x_2) + \dots \\ \dots + (x - x_0)(x - x_1) \dots (x - x_{n-1})f(x_0, x_1, \dots, x_n). \quad (4.13)$$

В случае равноотстоящих узлов интерполяции

$$x_1 = x_0 + h, \quad x_2 = x_0 + 2h, \quad \dots, \quad x_n = x_0 + nh$$

из формулы (4.13) с учётом равенств (4.11) получается *первая интерполяционная формула Ньютона*

$$f(x) \approx y_0 + \frac{\Delta y_0}{h}(x - x_0) + \frac{\Delta^2 y_0}{2!h^2}(x - x_0)(x - x_1) + \frac{\Delta^3 y_0}{3!h^3}(x - x_0)(x - x_1)(x - x_2) + \\ \dots + \frac{\Delta^n y_0}{n!h^n}(x - x_0)(x - x_1) \dots (x - x_{n-1}). \quad (4.14)$$

Эта формула применяется для интерполирования в начале отрезка интерполяции, когда значение $\frac{x - x_0}{h}$ мало по абсолютной величине.

Первую интерполяционную формулу Ньютона называют по этой причине *формулой для интерполирования вперёд*. За начальное значение x_0 можно принимать любое табличное значение аргумента x .

Когда значение аргумента находится ближе к концу отрезка интерполяции, применять первую интерполяционную формулу Ньютона становится невыгодно. В этом случае применяется *формула для интерполирования назад - вторая интерполяционная формула Ньютона*

$$f(x) \approx y_n + \frac{\Delta y_{n-1}}{h}(x - x_n) + \frac{\Delta^2 y_{n-2}}{2!h^2}(x - x_n)(x - x_{n-1}) + \frac{\Delta^3 y_{n-3}}{3!h^3}(x - x_n) \cdot \\ \cdot (x - x_{n-1})(x - x_{n-2}) + \dots + \frac{\Delta^n y_0}{n!h^n}(x - x_n)(x - x_{n-1}) \dots (x - x_1). \quad (4.15)$$

В формуле (4.14) коэффициенты многочлена содержат конечные разности различных порядков, принадлежащие верхней (нисходящей) строке таблицы разностей (см. таблицу на с.28). В формуле (2.15) коэффициенты многочлена содержат конечные разности различных порядков, принадлежащие нижней (восходящей) строке этой таблицы.

Погрешность интерполяционных формул Ньютона (4.14) и (4.15) оценивается неравенствами:

$$|f(x) - P_n(x)| \leq \frac{h^{n+1} M_{n+1}}{(n+1)!} |t(t-1)(t-2) \cdot \dots \cdot (t-n)|,$$

$$|f(x) - P_n(x)| \leq \frac{h^{n+1} M_{n+1}}{(n+1)!} |t(t+1)(t+2) \cdot \dots \cdot (t+n)|,$$

где $M_{n+1} = \max_{x_0 \leq x \leq x_n} |f^{(n+1)}(x)|$, $t = \frac{x - x_0}{h}$.

Пример 6. Найти интерполяционный многочлен Ньютона $P(x)$, для которого $P(1) = 0$, $P(3) = 6$, $P(6) = 18$.

Решение. В данном случае $x_0 = 1$, $x_1 = 3$, $x_2 = 6$,
 $y_0 = 0$, $y_1 = 6$, $y_2 = 18$.

Заметим, что узлы не являются равноотстоящими, т. к. $x_1 - x_0 \neq x_2 - x_1$.

При $n = 2$ формула (2.12) принимает вид

$$P_2(x) = y_0 + (x - x_0) f(x_0, x_1) + (x - x_0)(x - x_1) f(x_0, x_1, x_2).$$

Вычислим разделённые разности:

$$f(x_0, x_1) = \frac{y_1 - y_0}{x_1 - x_0} = \frac{6 - 0}{3 - 1} = \frac{6}{2} = 3,$$

$$f(x_2, x_1) = \frac{y_2 - y_1}{x_2 - x_1} = \frac{18 - 6}{6 - 3} = \frac{12}{3} = 4,$$

$$f(x_2, x_1, x_0) = \frac{f(x_2, x_1) - f(x_1, x_0)}{x_2 - x_0} = \frac{4 - 3}{6 - 1} = \frac{1}{5}.$$

Искомый интерполяционный многочлен Ньютона

$$P_2(x) = 3(x - 1) + 0,2(x - 1)(x - 3).$$

Раскрыв скобки и приведя подобные члены, получим многочлен в каноническом виде:

$$P_2(x) = 0,2(x^2 + 11x - 12) = 0,2x^2 + 2,2x - 2,4.$$

Пример 7. Найти интерполяционные многочлены Ньютона «интерполирования вперёд» и «интерполирования назад» для функции, заданной таблицей

x	0	1	2	3
y	2	1	4	10

Найти значение функции при $x = 0,5$ и $x = 2,5$.

Решение. В данном случае $x_0 = 0$, $x_1 = 1$, $x_2 = 2$, $x_3 = 3$,

$$y_0 = 2, \quad y_1 = 1, \quad y_2 = 4, \quad y_3 = 10.$$

Узлы интерполяции являются равноотстоящими:

$$x_1 - x_0 = x_2 - x_1 = x_3 - x_2 = 1.$$

Составим таблицу конечных разностей.

x	y	Δy	$\Delta^2 y$	$\Delta^3 y$
0	<u>2</u>			
1	1	<u>-1</u>		
2	4	3	<u>4</u>	
3	10	6	3	<u>-1</u>

Числа верхней (нисходящей) строки этой таблицы (подчёркнуты одной чертой) входят множителями в коэффициенты многочлена Ньютона для формулы «интерполирования вперёд».

При $n = 3$ правая часть формулы (2.14) принимает вид:

$$P_3^e(x) = \\ = \\ y_0 + \frac{\Delta y_0}{h}(x-x_0) + \frac{\Delta^2 y_0}{2!h^2}(x-x_0)(x-x_1) + \frac{\Delta^3 y_0}{3!h^3}(x-x_0)(x-x_1)(x-x_2).$$

Подставив в формулу условия задачи и указанные числа, получим

$$P_3^e(x) = 2 + (-\frac{1}{1!})(x-0) + (\frac{4}{2!})(x-0)(x-1) + (-\frac{1}{3!})(x-0)(x-1)(x-2) = \\ = 2 - x + 2x(x-1) - \frac{1}{6}x(x-1)(x-2).$$

Числа нижней (восходящей) строки этой таблицы (выделены жирным шрифтом) входят множителями в коэффициенты многочлена Ньютона для формулы «интерполирования назад».

При $n = 3$ правая часть формулы (2.15) принимает вид

$$P_3^H(x) = y_3 + \frac{\Delta y_2}{h}(x-x_3) + \frac{\Delta^2 y_1}{2!h^2}(x-x_3)(x-x_2) + \\ + \frac{\Delta^3 y_0}{3!h^3}(x-x_3)(x-x_2)(x-x_1).$$

Подставив в формулу условия задачи и указанные числа, получим

$$P_3^H(x) = 10 + 6(x-3) + \frac{3}{2}(x-3)(x-2) - \frac{1}{6}(x-3)(x-2)(x-1).$$

Найденные многочлены $P_3^e(x)$ и $P_3^H(x)$ отличаются только формой. Раскрыв скобки и приведя подобные члены, получим один и тот же многочлен в каноническом виде:

$$P_3(x) = -\frac{1}{6}x^3 + \frac{5}{2}x^2 - \frac{10}{3}x + 2.$$

Значение функции при $x = 0,5$ найдём с помощью первого многочлена:

$$f(0,5) = P_3^6(0,5) = 2 - 0,5 + 2 \cdot 0,5 \cdot (0,5 - 1) - \frac{1}{6} \cdot 0,5 \cdot (0,5 - 1)(0,5 - 2) = 1,5 - 0,5 - \frac{1}{16} = 1 - \frac{1}{16} = \frac{15}{16}.$$

Значение функции при $x = 2,5$ найдём с помощью второго многочлена:

$$f(2,5) = P_3^i(2,5) = 10 + 6(2,5 - 3) + \frac{3}{2}(2,5 - 3)(2,5 - 2) - \frac{1}{6}(2,5 - 3)(2,5 - 2)(2,5 - 1) = 10 - 3 - \frac{3}{8} + \frac{1}{16} = 6\frac{11}{16}.$$

Лабораторная работа № 4

Интерполирование функций

Цель: изучить интерполяционные многочлены Лагранжа и Ньютона, научиться строить их.

Задание 1. Найти интерполяционный многочлен Ньютона $P(x)$

для функции $f(x) = \frac{2^{x-1}}{n}$, где n - номер варианта, по её значениям в точках: $x_0 = -1$, $x_1 = 0$, $x_2 = 1$, $x_3 = 2$, $x_4 = 3$.

Вычислить значения $P(-0.5)$, $P(-0.21)$, $P(1.29)$, $P(2.56)$. Оценить погрешность вычислений.

Задание 2. Для функции $f(x)$ найти интерполяционный многочлен Лагранжа второго порядка, выбрав систему трёх равномерно расположенных на отрезке $[a, b]$ узлов. Оценить погрешность интерполяции в точке ξ .

Вариант	f(x)	a ≤ x ≤ b	ξ
1	cos x ²	0 ≤ x ≤ π / 4	π / 12
2	x sin x	0 ≤ x ≤ π / 4	π / 10
3	x cos x	0 ≤ x ≤ π / 6	π / 14

4	$\sin x^2$	$0 \leq x \leq \pi / 8$	$\pi / 12$
5	$\cos (3x)^2$	$0 \leq x \leq \pi / 2$	$\pi / 12$
6	$2x \sin (x/2)$	$0 \leq x \leq \pi / 4$	$\pi / 9$
7	$3x \cos (x/3)$	$0 \leq x \leq \pi / 6$	$\pi / 10$
8	e^{2x}	$0 \leq x \leq 0,6$	0,3
9	$\sin 4x$	$0 \leq x \leq \pi / 4$	$\pi / 32$
10	e^{-3x}	$0 \leq x \leq 0,4$	0,1
11	$\cos 3x$	$0 \leq x \leq \pi / 6$	$\pi / 8$
12	$\operatorname{tg} 2x$	$-\pi / 6 \leq x \leq \pi / 6$	$\pi / 7$
13	$\sin (2x)^2$	$0 \leq x \leq \pi / 8$	$\pi / 12$
14	e^{-x^2}	$0 \leq x \leq 0,4$	0,2
15	$\operatorname{tg} x^2$	$-\pi / 3 \leq x \leq \pi / 3$	$\pi / 5$

Задание 4. Функция $f(x)$ определена на отрезке $[1; 1,2]$.

Построить многочлен Лагранжа, интерполирующий функцию $f(x)$ на этом отрезке по системе трёх равномерно расположенных узлов.

Найти значения многочлена Лагранжа в точках 1,05; 1,09; 1,12; 1,15; 1,17. Полученные результаты сравнить с табличными значениями и дать оценку точности интерполяции.

Функцию $f(x)$ и систему трёх равномерно расположенных узлов из отрезка $[1; 1,2]$ определяет преподаватель (по таблице).

x	e^x	e^{-x}	$\sin x$	$\cos x$	$\ln x$
1,00	2,7183	0,3679	0,8415	0,5403	0

01	2,7456	0,3642	0,8468	0,5319	0,01
02	2,7732	0,3606	0,8521	0,5234	0,0198
03	2,8011	0,3570	0,8573	0,5148	0,0296
04	2,8292	0,3535	0,8624	0,5062	0,0392
05	2,8577	0,3499	0,8674	0,4976	0,0488
06	2,8864	0,3465	0,8724	0,4889	0,0583
07	2,9154	0,3430	0,8772	0,4801	0,0677
08	2,9447	0,3396	0,8820	0,4713	0,0770
09	2,9743	0,3362	0,8866	0,4625	0,0862
1,10	3,0042	0,3329	0,8912	0,4536	0,0953
11	3,0344	0,3296	0,8957	0,4447	0,1044
12	3,0649	0,3263	0,9001	0,4357	0,1133
13	3,0957	0,3230	0,9044	0,4267	0,1222
14	3,1268	0,3198	0,9086	0,4176	0,1310
15	3,1582	0,3166	0,9128	0,4085	0,1398
16	3,1899	0,3135	0,9168	0,3993	0,1484
17	3,2220	0,3104	0,9208	0,3902	0,1570
18	3,2544	0,3073	0,9246	0,3809	0,1655
19	3,2871	0,3042	0,9284	0,3717	0,1740
1,20	3,3201	0,3012	0,9320	0,3624	0,1823

На рис. 30-31 приведены примеры решения задач при помощи MathCAD.

Пример 1. Найти интерполяционный многочлен Лагранж $P(x)$ для функции, заданной таблицей

$$x_0 := 0 \quad x_1 := 1 \quad x_2 := 2 \quad x_3 := 3$$

$$y_0 := 6 \quad y_1 := 0 \quad y_2 := -10 \quad y_3 := -30$$

$$P_0(x) := y_0 \cdot \frac{(x - x_1)}{(x_0 - x_1)} \cdot \frac{(x - x_2)}{(x_0 - x_2)} \cdot \frac{(x - x_3)}{(x_0 - x_3)}$$

$$P_1(x) := y_1 \cdot \frac{(x - x_0)}{(x_1 - x_0)} \cdot \frac{(x - x_2)}{(x_1 - x_2)} \cdot \frac{(x - x_3)}{(x_1 - x_3)}$$

$$P_2(x) := y_2 \cdot \frac{(x - x_1)}{(x_2 - x_1)} \cdot \frac{(x - x_0)}{(x_2 - x_0)} \cdot \frac{(x - x_3)}{(x_2 - x_3)}$$

$$P_3(x) := y_3 \cdot \frac{(x - x_1)}{(x_3 - x_1)} \cdot \frac{(x - x_2)}{(x_3 - x_2)} \cdot \frac{(x - x_0)}{(x_3 - x_0)}$$

$$P(x) := P_0(x) + P_1(x) + P_2(x) + P_3(x)$$

$$P(x) \text{ simplify } \rightarrow -x^3 + x^2 - 6 \cdot x + 6$$

$$x := 0, 1.. 3$$

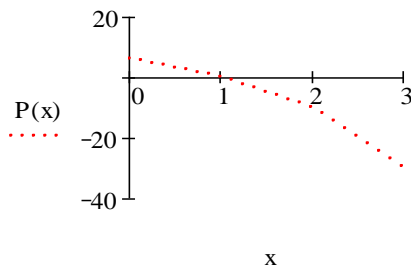


Рис. 30

Пример 2 Найти интерполяционный многочлен Лагранжа $P(x)$ для функции $y=\cos(x)$ на отрезке $[0;\pi/4]$, если заданы значения функции в трех узлах интерполяции. Вычислить приближенное значение функции $\cos(\pi/12)$ и оценить погрешность результата вычислений

$$x_0 := 0 \quad x_1 := \frac{\pi}{6} \quad x_2 := \frac{\pi}{4} \quad y_0 := 1 \quad y_1 := \frac{\sqrt{3}}{2} \quad y_2 := \frac{\sqrt{2}}{2}$$

$$P_0(x) := y_0 \cdot \frac{(x-x_1)(x-x_2)}{(x_0-x_1)(x_0-x_2)} \quad P_1(x) := y_1 \cdot \frac{(x-x_0)(x-x_2)}{(x_1-x_0)(x_1-x_2)}$$

$$P_2(x) := y_2 \cdot \frac{(x-x_0)(x-x_1)}{(x_2-x_0)(x_2-x_1)} \quad P(x) := P_0(x) + P_1(x) + P_2(x)$$

$$P(x) \rightarrow 24 \cdot \frac{\left(x - \frac{1}{6} \cdot \pi\right)^2}{\pi^2} \cdot \left(x - \frac{1}{4} \cdot \pi\right) - 36 \cdot \sqrt{3} \cdot \frac{x}{\pi} \cdot \left(x - \frac{1}{4} \cdot \pi\right) + 24 \cdot \sqrt{2} \cdot \frac{\left(x - \frac{1}{6} \cdot \pi\right)}{\pi} \cdot x$$

$$P\left(\frac{\pi}{12}\right) = 0.96366 \quad \left| \cos\left(\frac{\pi}{12}\right) - P\left(\frac{\pi}{12}\right) \right| = 2.269 \times 10^{-3}$$

$$R_3(x) := (x-x_0) \cdot (x-x_1) \cdot (x-x_2) \quad R_3\left(\frac{\pi}{12}\right) = 0.036$$

$$\frac{d^3}{dx^3} \cos(x) \rightarrow \sin(x) \quad \sin\left(\frac{\pi}{4}\right) = 0.707$$

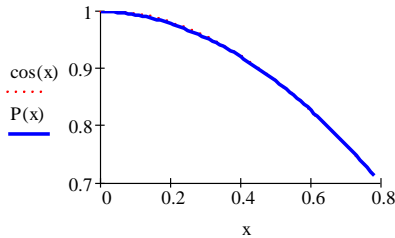
$$\sin(0) = 0 \quad \sin\left(\frac{\pi}{6}\right) = 0.5 \quad M_3 := \sin\left(\frac{\pi}{4}\right)$$

$$\frac{M_3}{3!} \cdot \left| R_3\left(\frac{\pi}{12}\right) \right| = 4.229 \times 10^{-3}$$

Таким образом, вычисленное значение отличается от истинного на 0.004229

$$\text{Вычисленное значение} \quad \cos\left(\frac{\pi}{12}\right) = 0.96593$$

$$x := 0, 0.01, \dots, \frac{\pi}{4}$$



Таким образом, многочлен Лагранжа почти полностью совпал с функцией $y=\cos x$

Рис. 31. Продолжение решения

Построим интерполяционный многочлен Ньютона с помощью метода неопределенных коэффициентов. Для этого запишем искомый многочлен в виде

$$P_n(x) = b_0 + b_1(x - x_0) + b_2(x - x_0)(x - x_1) + b_3(x - x_0)(x - x_1)(x - x_2) + \dots + b_n(x - x_0)\dots(x - x_{n-1}).$$

Последовательно подставляя в формулу вместо x данные значения x_0, x_1, \dots, x_{n+1} , получим для нахождения неопределенных коэффициентов b_0, b_1, \dots, b_n «треугольную» систему

$$\begin{cases} y_0 = b_0, \\ y_1 = b_0 + b_1(x_1 - x_0), \\ y_2 = b_0 + b_1(x_2 - x_0) + b_2(x_2 - x_0)(x_2 - x_1), \\ \dots, \\ y_{n+1} = b_0 + b_1(x_{n+1} - x_0) + \dots + b_n(x_{n+1} - x_0) \cdot \dots \cdot (x_{n+1} - x_n) \end{cases}$$

уравнений (при подстановке в равенство вместо x числа x_0 в правой части равенства обратились в нуль все слагаемые, кроме первого: там везде был множитель $(x - x_0)$, обратившийся в нуль; при подстановке $x = x_1$ обратились в нуль все слагаемые, кроме первого и второго – они содержат множитель $(x - x_1)$ и т.д.).

Полученную систему удобно решать:

из первого её уравнения находим свободный член искомого многочлена b_0 ;

подставив его во второе уравнение, находим коэффициент b_1

при $b_1 = \frac{y_1 - b_0}{x_1 - x_0}$ первой степени x в искомом многочлене т.д.

Для интерполяционного многочлена Ньютона можно выписать явные выражения коэффициентов через данные задачи, а также и оценки точности замены неизвестной функции $f(x)$ этим многочленом. Рассмотрим построение многочлена Ньютона методом неопределенных коэффициентов.

Пример Найти интерполяционный многочлен Ньютона $P_1(x)$ для функции, заданной таблицей

$$\begin{array}{cccc} x_0 := 0 & x_1 := 1 & x_2 := 2 & x_3 := 3 \\ y_0 := 6 & y_1 := 0 & y_2 := -10 & y_3 := -30 \end{array}$$

Найдем коэффициенты b_0, b_1, b_2

$$b_0 := y_0 \quad y_0 = 6$$

$$y_1 = b_0 + b_1 \cdot (x_1 - x_0) \rightarrow 0 = 6 + b_1$$

$$0 = 6 + b_1 \quad b_1 := -6$$

$$y_2 = b_0 + b_1 \cdot (x_2 - x_0) + b_2 \cdot (x_2 - x_0) \cdot (x_2 - x_1) \rightarrow -10 = -6 + 2 \cdot b_2$$

$$-10 = -6 + 2 \cdot b_2$$

$$b_2 := -2$$

Найдем коэффициент b_3 и построим многочлен Ньютона

$$y_3 = b_0 + b_1 \cdot (x_3 - x_0) + b_2 \cdot (x_3 - x_0)(x_3 - x_1) + b_3 \cdot (x_3 - x_0)(x_3 - x_1)(x_3 - x_2)$$

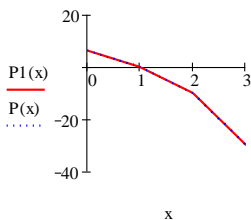
$$b_0 + b_1 \cdot (x_3 - x_0) + b_2 \cdot (x_3 - x_0) \cdot (x_3 - x_1) + b_3 \cdot (x_3 - x_0) \cdot (x_3 - x_1) \cdot (x_3 - x_2) \rightarrow -24 + 6 \cdot b_3$$

$$-30 = -24 + 6 \cdot b_3 \quad b_3 := -1$$

$$x := 0, 1..3$$

$$P_1(x) := 6 - 6 \cdot (x - x_0) - 2 \cdot (x - x_0)(x - x_1) - (x - x_0)(x - x_1)(x - x_2) = x^3 + x^2 - 6 \cdot x + 6$$

$$P_1(x) \rightarrow -6 \cdot x + 6 - 2 \cdot x \cdot (x - 1) - x \cdot (x - 1) \cdot (x - 2) \text{ simplify} \rightarrow -x^3 + x^2 - 6 \cdot x + 6$$



Как видим, многочлен Ньютона совпал с многочленом Лагранжа для данных узлов

Рис. 32

В пакете прикладных программ *Mathcad* имеются встроенные функции, которые позволяют строить через экспериментальные точки кривые разной степени гладкости.

Наиболее простым методом интерполяции есть линейная. При линейной интерполяции аппроксимирующая функция соединяет экспериментальные точки отрезками прямых линий.

Для линейной интерполяции используется функция $linterp(x,y,n)$.

Где

x - вектор экспериментальных значений аргументов, расположенных в порядке возрастания;

y - вектор значений функций, данных по условию;

n - значение аргумента, при котором вычисляется значение функции.

В большинстве случаев желательно соединять экспериментальные точки не отрезками прямой линии, а гладкой кривой.

В этом случае используется сплайн - интерполяция.

4.1.3 Сплайн-интерполяция

Пример.

Рассмотрим пример использования методов интерполяции для описания зависимости коэффициента теплоотдачи от цилиндра, при поперечном обтекании его потоком жидкости.

Значение критерия Нуссельта Nu_d в зависимости от угловой координаты при поперечном обтекании цилиндра (значение критерия Рейнольдса $Re_d = 70800$, определяющий размер - диаметр цилиндра d) приведены в таблице.

Угловая координата	0	20	40	60	80	100	120	140	160	180
Число Нуссельта	260	255	245	205	100	115	145	165	185	200

$$M1 := \begin{pmatrix} 0 & 260 \\ 20 & 255 \\ 40 & 245 \\ 60 & 205 \\ 80 & 100 \\ 100 & 115 \\ 120 & 145 \\ 140 & 165 \\ 160 & 185 \\ 180 & 200 \end{pmatrix}$$
Массив данных

$n := 0, 0.1 \dots \max(M1 \langle 0 \rangle)$ **Задание точек, в торых с шагом 0,1 будет интерполироваться линейная функция**

$Yline(n) := \text{linterp}(M1 \langle 0 \rangle, M1 \langle 1 \rangle, n)$

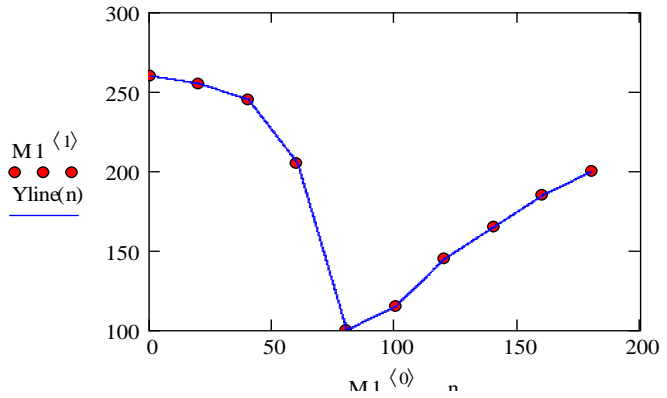
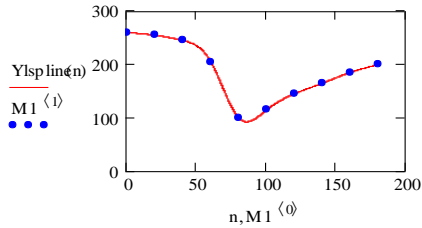


Рис. 33 Линейная интерполяция базы данных

Сравнение графиков на рис. 34 позволяет сделать вывод, что интерполяционные кривые, создаваемых функциями *lspline(x,y)*, *psplsne(x,y)* и *cspline(x,y)*, не имеют значительных погрешностей.

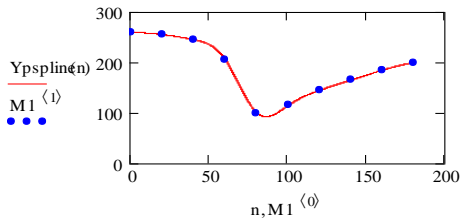
Интерполяция сплайном lspline(n)

$$Y_{lspline}(n) := \text{interp}(lspline(M1^{(0)}, M1^{(1)}), M1^{(0)}, M1^{(1)}, n)$$

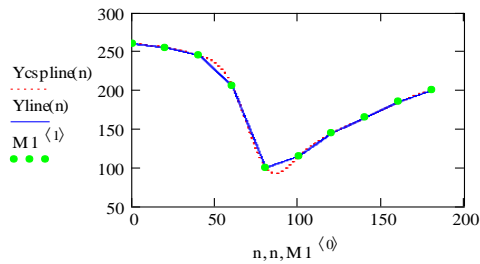


Интерполяция сплайном pspline(n)

$$Y_{pspline}(n) := \text{interp}(pspline(M1^{(0)}, M1^{(1)}), M1^{(0)}, M1^{(1)}, n)$$



$$Y_{cspline}(n) := \text{interp}(cspline(M1^{(0)}, M1^{(1)}), M1^{(0)}, M1^{(1)}, n)$$



Интерполяция сплайном cspline(n) и линейная интерполяция Yline(n) в сравнении

Кубическая сплайн-интерполяция позволяет провести кривую через набор точек таким образом, чтобы первые и вторые производные кривой были непрерывны в каждой точке. Эта кривая образуется путем создания ряда кубических полиномов, проходящих через наборы из трех смежных точек. Кубические полиномы затем состыковываются друг с другом, чтобы образовать одну кривую.

Для кубической сплайновой интерполяции в пакете прикладных программ используется функция $\text{interp}(vs,x,y,n)$. Где vs - вектор второй производной, созданной функциями $\text{lspline}(x,y)$, $\text{psplsne}(x,y)$ или $\text{cspline}(x,y)$.

$\text{lspline}(x,y)$ - создает вектор коэффициентов кривой, которая приближается к прямой линии в предельных точках;

$\text{psplsne}(x,y)$ - создает вектор коэффициентов кривой, которая приближается к квадратичной параболе в предельных точках;

$\text{cspline}(x,y)$ - создает вектор коэффициентов кривой, которая приближается к кубической параболе в предельных точках.

Вопросы для самоконтроля для самоконтроля

1. Что называется интерполяцией? Сформулируйте задачу интерполирования функции.
2. Что называется интерполяционным многочленом Лагранжа? Как он находится?
3. Что называется интерполяционным многочленом Ньютона? Как он находится?
4. Как оценить погрешность вычислений?
5. В чем преимущества сплайн-интерполяции по сравнению с интерполяционными полиномами?
6. Какая функция MathCAD реализует линейную интерполяцию?
7. Какие функции кубической сплайн-интерполяции вам известны, охарактеризуйте последовательность из использования?

ГЛАВА 5

АППРОКСИМАЦИЯ ФУНКЦИЙ

5.1. Метод наименьших квадратов

Пусть в результате эксперимента получена таблица значений x и y

x	x_1	x_2	x_3	\dots	x_n
$f(x)$	y_1	y_2	y_3	\dots	y_n

Требуется найти функцию вида $y = F(x)$, которая в точках x_1, x_2, \dots, x_n принимает значения, наиболее близкие к табличным значениям y_1, y_2, \dots, y_n . Такую формулу называют *эмпирической формулой* или *уравнением регрессии* y на x , а саму функцию называют *приближающей* или *аппроксимирующей функцией*.

На практике эту аппроксимирующую функцию $F(x)$ находят следующим образом. По таблице строят точечный график функции f , по которому устанавливают вид приближающей функции из числа известных.

В качестве приближающей функции $y = F(x)$ в зависимости от характера точечного графика часто используют следующие элементарные функции:

$$\begin{aligned}
 & y = ax + b; & y = ax^2 + bx + c; & y = a \cdot x^m; \\
 & y = b \cdot a^x; & y = a + b \cdot \sin x; & y = a \cdot \ln x + b; \\
 & y = \frac{1}{ax + b}; & y = a \cdot \frac{1}{x} + b; & y = \frac{x}{ax + b}; & y = a \cdot e^{mx},
 \end{aligned}$$

где a, b, c, m - некоторые параметры.

Таким образом, если вид приближающей функции установлен, то задача сводится к отысканию значений параметров. Их можно вычислить по методу наименьших квадратов, суть которого заключается в следующем.

Пусть требуется найти аппроксимирующую функцию, например с тремя параметрами:

$$y = F(x, a, b, c). \quad (5.1)$$

Для x_i (где $i = 1, 2, \dots, n$) из таблицы эта функция примет значения

$$\bar{y}_i = F(x_i, a, b, c),$$

которые должны как можно меньше отличаться от заданных (табличных) значений y_i , т.е. разность $\bar{y}_i - y_i$ должна быть близка к нулю. Поэтому сумма квадратов разностей соответствующих значений функций $f(x)$ и $F(x)$

$$\sum_{i=1}^n (y_i - F(x_i, a, b, c))^2 = \Phi(a, b, c)$$

также должна принимать минимальное значение.

Таким образом, задачу свели к отысканию минимума функции $\Phi(a, b, c)$. Используем необходимое условие экстремума:

$$\begin{cases} \frac{\partial \Phi}{\partial a} = 0, \\ \frac{\partial \Phi}{\partial b} = 0, \\ \frac{\partial \Phi}{\partial c} = 0, \end{cases} \quad \text{или} \quad \begin{cases} \sum_{i=1}^n (y_i - F(x_i, a, b, c)) \cdot F'_a(x_i, a, b, c) = 0; \\ \sum_{i=1}^n (y_i - F(x_i, a, b, c)) \cdot F'_b(x_i, a, b, c) = 0; \\ \sum_{i=1}^n (y_i - F(x_i, a, b, c)) \cdot F'_c(x_i, a, b, c) = 0. \end{cases} \quad (5.2)$$

Решив эту систему из трёх уравнений с тремя неизвестными, получим значения параметров a , b и c - следовательно, получим конкретный вид приближающей функции $F(x, a, b, c)$.

Очевидно, что значения найденной функции $F(x, a, b, c)$ в точках x_1, x_2, \dots, x_n будут отличаться от табличных значений y_1, y_2, \dots, y_n .

Значения разностей $\varepsilon_i = y_i - F(x_i, a, b, c)$, где $i = 1, 2, \dots, n$, называются *отклонениями* данных значений y от вычисленных.

Сумма квадратов отклонений $\sigma = \sum_{i=1}^n \varepsilon_i^2$ должна быть наименьшей. Отметим, что из нескольких приближений для одной и той же табличной функции лучшим является то, для которого σ имеет наименьшее значение. В нашем случае приближающая функция зависела от трёх параметров, однако изменение количества параметров повлияет только на изменение количества уравнений системы (5.2), а суть метода останется прежней.

Рассмотрим частные случаи нахождения эмпирических формул.

5.1.1 Линейная регрессия

Пусть требуется найти приближающую функцию в виде линейной

$$F(x, a, b) = ax + b.$$

Так как её частные производные по параметрам a и b : $F'_a(x, a, b) = x$, $F'_b(x, a, b) = 1$, то система (5.2) примет вид:

$$\begin{cases} \sum_{i=1}^n (y_i - ax_i - b) \cdot x_i = 0; \\ \sum_{i=1}^n (y_i - ax_i - b) \cdot 1 = 0. \end{cases}$$

После несложных преобразований её можно привести к виду:

$$\begin{cases} \sum_{i=1}^n x_i y_i = a \sum_{i=1}^n x_i^2 + b \sum_{i=1}^n x_i; \\ \sum_{i=1}^n y_i = a \sum_{i=1}^n x_i + n \cdot b. \end{cases}$$

Решив систему, получим значения параметров a и b , следовательно, и конкретный вид приближающей функции

$$F(x, a, b) = ax + b.$$

Заметим, что в пакете MathCAD решать систему уравнений удобнее в матричном виде. Так, в случае линейной регрессии параметры a и b легко найти из матричного уравнения

$$\begin{pmatrix} a \\ b \end{pmatrix} = \begin{pmatrix} \sum_{i=1}^n x_i^2 & \sum_{i=1}^n x_i \\ \sum_{i=1}^n x_i & n \end{pmatrix}^{-1} \cdot \begin{pmatrix} \sum_{i=1}^n x_i y_i \\ \sum_{i=1}^n y_i \end{pmatrix}.$$

5.1.2 Квадратичная регрессия

Пусть требуется найти приближающую функцию в виде квадратичной

$$F(x, a, b, c) = ax^2 + bx + c.$$

Так как её частные производные по параметрам a , b и c соответственно равны:

$$F'_a(x, a, b, c) = x^2,$$

$$F'_b(x, a, b, c) = x,$$

$$F'_c(x, a, b, c) = 1,$$

то система (5.2) примет вид:

$$\begin{cases} \sum_{i=1}^n (y_i - ax_i^2 - bx_i - c)x_i^2 = 0; \\ \sum_{i=1}^n (y_i - ax_i^2 - bx_i - c) \cdot x_i = 0; \\ \sum_{i=1}^n (y_i - ax_i^2 - bx_i - c) \cdot 1 = 0. \end{cases}$$

Как и в предыдущем случае, параметры a , b , c легко находятся из матричного уравнения

$$\begin{pmatrix} a \\ b \\ c \end{pmatrix} = \begin{pmatrix} \sum_{i=1}^n x_i^4 & \sum_{i=1}^n x_i^3 & \sum_{i=1}^n x_i^2 \\ \sum_{i=1}^n x_i^3 & \sum_{i=1}^n x_i^2 & \sum_{i=1}^n x_i \\ \sum_{i=1}^n x_i^2 & \sum_{i=1}^n x_i & n \end{pmatrix}^{-1} \cdot \begin{pmatrix} \sum_{i=1}^n y_i x_i^2 \\ \sum_{i=1}^n y_i x_i \\ \sum_{i=1}^n y_i \end{pmatrix}.$$

5.1.3 Геометрическая регрессия

Пусть требуется найти приближающую функцию в виде степенной

$$F(x, a, m) = a \cdot x^m.$$

При условии, что $a > 0$ и в заданной таблице значения аргумента и значения функции положительны, прологарифмируем равенство

$$\ln F = \ln a + m \cdot \ln x.$$

Введём следующие обозначения:

$$u = \ln x; \quad A = m; \quad B = \ln a,$$

тогда $\ln F$ будет функцией от u : $\Phi(u, A, B) = Au + B$.

Таким образом, нахождение параметров степенной функции мы свели к нахождению параметров линейной функции. Поэтому дальнейшее решение поставленной задачи будет аналогично первому случаю.

Так как частные производные функции $\Phi(u, A, B)$ по параметрам A и B :

$$\Phi'_A = u, \quad \Phi'_B = 1,$$

то система (5.2) примет вид:

$$\begin{cases} \sum_{i=1}^n u_i \cdot y_i = A \cdot \sum_{i=1}^n u_i^2 + B \cdot \sum_{i=1}^n u_i; \\ \sum_{i=1}^n y_i = A \cdot \sum_{i=1}^n u_i + B \cdot n. \end{cases}$$

Решив систему, получим значения параметров A и B линейной функции $\Phi(u, A, B)$. Сделав обратную замену: $a = e^B$, $m = A$, получим требуемые параметры степенной функции $F(x, a, m) = a \cdot x^m$.

5.1.4 Тригонометрическая функция

Пусть требуется найти приближающую функцию в виде тригонометрической $F(x, a, b) = a + b \cdot \sin x$.

Поскольку её частные производные по параметрам a и b равны:

$$F'_a(x, a, b) = 1; \quad F'_b(x, a, b) = \sin x,$$

то система (5.2) примет вид:

$$\begin{cases} \sum_{i=1}^n (y_i - a - b \cdot \sin x_i) = 0; \\ \sum_{i=1}^n (y_i - a - b \cdot \sin x_i) \cdot \sin x_i = 0, \end{cases} \quad \text{или}$$

$$\begin{cases} \sum_{i=1}^n y_i = n \cdot a + b \cdot \sum_{i=1}^n \sin x_i ; \\ \sum_{i=1}^n y_i \cdot \sin x_i = a \cdot \sum_{i=1}^n \sin x_i + b \cdot \sum_{i=1}^n \sin^2 x_i . \end{cases}$$

Откуда, после элементарных преобразований, можно получить требуемые параметры.

Если решать систему матричным способом, то уравнение для вычисления параметров a и b будет следующим:

$$\begin{pmatrix} a \\ b \end{pmatrix} = \begin{pmatrix} n & \sum_{i=1}^n \sin x_i \\ \sum_{i=1}^n \sin x_i & \sum_{i=1}^n \sin^2 x_i \end{pmatrix}^{-1} \cdot \begin{pmatrix} \sum_{i=1}^n y_i \\ \sum_{i=1}^n y_i \cdot \sin x_i \end{pmatrix} .$$

5.1.5 Логарифмическая функция

Пусть требуется найти приближающую функцию в виде логарифмической $F(x, a, b) = a \cdot \ln x + b$.

Так как частные производные функции $F(x, a, b)$ по параметрам a и b равны:

$$F'_a(x, a, b) = \ln x ; \quad F'_b(x, a, b) = 1 ,$$

то система (5.2) примет вид:

$$\begin{cases} \sum_{i=1}^n (y_i - a \cdot \ln x_i - b) \cdot \ln x_i = 0 ; \\ \sum_{i=1}^n (y_i - a \cdot \ln x_i - b) \cdot 1 = 0 . \end{cases}$$

Откуда получим:

$$\begin{cases} \sum_{i=1}^n y_i = n \cdot b + a \cdot \sum_{i=1}^n \ln x_i ; \\ \sum_{i=1}^n y_i \cdot \ln x_i = a \cdot \sum_{i=1}^n \ln^2 x_i + b \cdot \sum_{i=1}^n \ln x_i . \end{cases}$$

После элементарных преобразований найдём требуемые параметры.

Матричное уравнение для вычисления параметров a и b имеет вид:

$$\begin{pmatrix} a \\ b \end{pmatrix} = \begin{pmatrix} \sum_{i=1}^n \ln^2 x_i & \sum_{i=1}^n \ln x_i \\ \sum_{i=1}^n \ln x_i & n \end{pmatrix}^{-1} \cdot \begin{pmatrix} \sum_{i=1}^n y_i \cdot \ln x_i \\ \sum_{i=1}^n y_i \end{pmatrix}.$$

5.1.6 Дробно-линейная функция

Пусть требуется найти приближающую функцию в виде дробно-линейной $F(x, a, b) = \frac{x}{ax + b}$.

Так как $\frac{1}{F(x, a, b)} = \frac{ax + b}{x} = a + \frac{b}{x}$, то, заменив в исходной таблице значения x и y их обратными величинами $u = 1/x$, $z = \Phi(u, a, b) = 1/y = 1/F(x, a, b)$,

можно свести задачу к нахождению параметров линейной функции, зависящей от u :

$$\Phi(u, a, b) = a + bu.$$

Поэтому дальнейшее решение поставленной задачи будет аналогично первому случаю – случаю линейной регрессии.

Так как частные производные функции $\Phi(u, a, b)$ по параметрам a и b :

$$\Phi'_b = u, \quad \Phi'_a = 1,$$

то система (5.2) примет вид:

$$\begin{cases} \sum_{i=1}^n u_i \cdot z_i = b \cdot \sum_{i=1}^n u_i^2 + a \cdot \sum_{i=1}^n u_i; \\ \sum_{i=1}^n z_i = b \cdot \sum_{i=1}^n u_i + a \cdot n. \end{cases}$$

Решив систему, получим значения параметров a и b дробно-линейной функции $F(x, a, b) = \frac{x}{ax + b}$.

Матричное уравнение для вычисления параметров a и b имеет вид:

$$\begin{pmatrix} b \\ a \end{pmatrix} = \begin{pmatrix} \sum_{i=1}^n \left(\frac{1}{x_i}\right)^2 & \sum_{i=1}^n \left(\frac{1}{x_i}\right) \\ \sum_{i=1}^n \left(\frac{1}{x_i}\right) & n \end{pmatrix}^{-1} \cdot \begin{pmatrix} \sum_{i=1}^n \left(\frac{1}{x_i}\right) \cdot \left(\frac{1}{y_i}\right) \\ \sum_{i=1}^n \left(\frac{1}{y_i}\right) \end{pmatrix}.$$

Мы рассмотрели частные случаи нахождения эмпирических формул в виде линейной, квадратичной, степенной, тригонометрической, логарифмической и дробно-линейной функций, параметры других элементарных функций вычисляются аналогично.

Лабораторная работа № 5

Нахождение эмпирических формул по методу наименьших квадратов

Цель: изучить метод наименьших квадратов, построить различные эмпирические формулы

Задание. По заданной таблице значений переменных постройте с помощью метода наименьших квадратов различные эмпирические формулы, сравните качество полученных приближений и выберите из них наилучшее.

Указания к выполнению лабораторной работы

Для нахождения приближающей функции $y = F(x)$ выполните следующие действия.

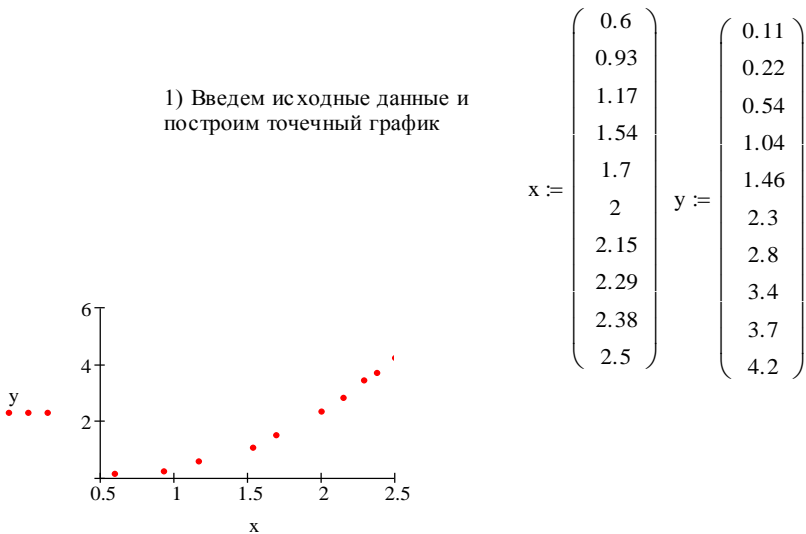
1. Постройте точечный график функции $y = f(x)$, где x , y - значения из таблицы.
2. Проведите плавную кривую, отражающую характер расположения точек. Сравните кривую, построенную по экспериментальным точкам, с образцами известных кривых (прямой, параболой, гиперболой, логарифмической кривой и другими). Установите вид приближающей функции.

3. Подберите параметры приближающей функции $y = F(x)$, используя метод наименьших квадратов.
4. Вычислите значение суммы квадратов отклонений

$$\sigma = \sum_{i=1}^n \varepsilon_i^2, \text{ где } \varepsilon_i = f(x_i) - F(x_i).$$

Замечание. Из нескольких приближений для одной и той же табличной функции (по принципу наименьших квадратов) лучшим является то, для которого σ имеет наименьшее значение.

Пример выполнения задания представлен на рис. 35-38.



2) Найдем параметры *линейной регрессии*

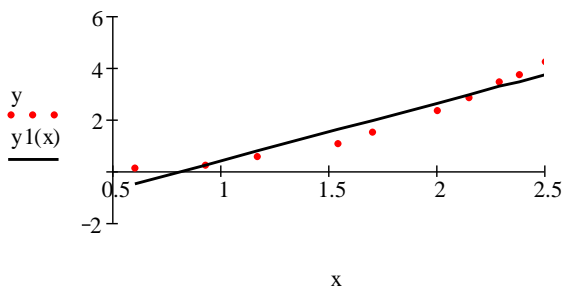
$$\begin{pmatrix} a \\ b \end{pmatrix} := \begin{pmatrix} x \cdot x & \sum x \\ \sum x & 10 \end{pmatrix}^{-1} \cdot \begin{pmatrix} x \cdot y \\ \sum y \end{pmatrix}$$

$a = 2.23 \quad b = -1.872$

$$y1(x) := a \cdot x + b$$

Рис. 35 (начало документа MathCAD)

3) Построим исходный график *линейный* график



4) Сравним исходные значения и значения *линейной* функции при заданных значениях аргумента

x =	0	y =	0	y1(x) =	0	y - y1(x) =	0	
	0	0.6	0	0.11	0	-0.534	0	0.644
	1	0.93	1	0.22	1	0.202	1	0.018
	2	1.17	2	0.54	2	0.737	2	-0.197
	3	1.54	3	1.04	3	1.562	3	-0.522
	4	1.7	4	1.46	4	1.919	4	-0.459
	5	2	5	2.3	5	2.588	5	-0.288
	6	2.15	6	2.8	6	2.922	6	-0.122
	7	2.29	7	3.4	7	3.235	7	0.165
	8	2.38	8	3.7	8	3.435	8	0.265
	9	2.5	9	4.2	9	3.703	9	0.497

5) Найдем погрешность первого приближения (остаточную дисперсию)

$$\varepsilon_1 := \left[\sum_i (y_i - y_1(x_i))^2 \right] \cdot \frac{1}{10} \quad \varepsilon_1 = 0.138$$

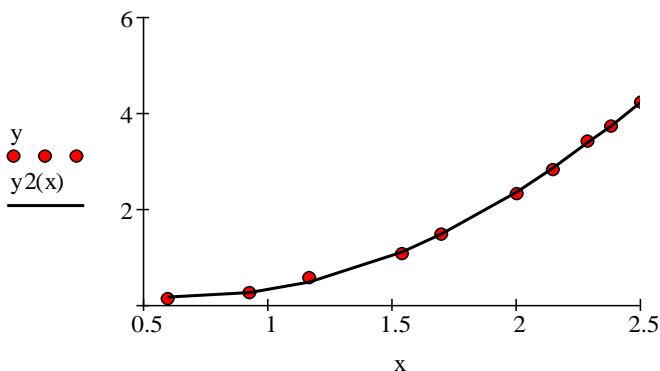
Рис. 36 (продолжение1)

$$\begin{pmatrix} a \\ b \\ c \end{pmatrix} := \begin{bmatrix} \sum_i (x_i)^4 & \sum_i (x_i)^3 & \sum_i (x_i)^2 \\ \sum_i (x_i)^3 & \sum_i (x_i)^2 & \sum_i x_i \\ \sum_i (x_i)^2 & \sum_i x_i & 10 \end{bmatrix}^{-1} \cdot \begin{bmatrix} \sum_i (x_i)^2 \cdot y_i \\ \sum_i x_i \cdot y_i \\ \sum_i y_i \end{bmatrix}$$

$$a = 1.17 \quad b = -1.477 \quad c = 0.591$$

$$y_2(x) := a \cdot x^2 + b \cdot x + c$$

7) Построим исходный график и график *квадратичной регрессии*



8) Найдем погрешность второго приближения

$$i := 0..9$$

$$\varepsilon_2 := \left[\sum_i (y_i - y_2(x_i))^2 \right] \cdot \frac{1}{10} \quad \varepsilon_2 = 1.275 \times 10^{-3}$$

Рис. 37 (продолжение 2)

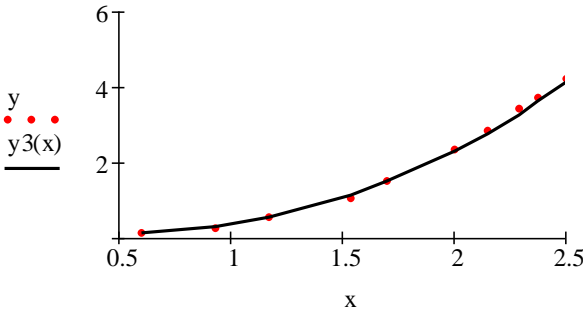
9) Найдем параметры *степенной регрессии*

$$\begin{pmatrix} m \\ N \end{pmatrix} := \begin{bmatrix} \sum_i (\ln(x_i))^2 & \sum_i \ln(x_i) \\ \sum_i \ln(x_i) & 10 \end{bmatrix}^{-1} \cdot \begin{pmatrix} \sum_i \ln(x_i) \cdot \ln(y_i) \\ \sum_i \ln(y_i) \end{pmatrix}$$

$m = 2.679$ $N = -1.043$ $n := e^N$

$$y_3(x) := n \cdot x^m$$

11) Построим исходный график *и степенной* график



12) Найдем погрешность третьего приближения

$$\varepsilon_3 := \left[\sum_i (y_i - y_3(x_i))^2 \right] \cdot \frac{1}{10} \quad \varepsilon_3 = 6.229 \times 10^{-3}$$

13) Сравним погрешности трех найденных приближений

$$\varepsilon_1 = 0.138 \quad \varepsilon_2 = 1.275 \times 10^{-3} \quad \varepsilon_3 = 6.229 \times 10^{-3}$$

Вывод: Наиболее подходящей приближающей функцией является квадратичная

Рис. 38

Вариант 1

0,7	0,93	1,17	1,54	1,7	2	2,15	2,29	2,38	2,5
0,19	0,32	0,64	1,04	1,46	2,54	3,01	3,74	4,2	4,96

Вариант 2

4,36	3,4	2,95	2,66	2,12	1,96	1,06	0,96	0,76	0,16
6,23	10,9	13,9	18,31	21,9	2	32,2	3	36,6	49,45
					4,1		3,4		

Вариант 3

2,21	3,82	4,43	5,34	5,84	6,19	6,29	7	8	9
52,43	58,6	60,7	67,31	70,5	73,3	75,3	80,1	91,52	105

Вариант 4

2,44	2,51	2,6	2,62	2,69	2,95	2,98	3,01	3,2	3,37
10,3	10,4	11,84	11,9	12,5	13,4	14,74	15,7	16,24	17,7

Вариант 5

51,7	52,8	55,7	58,9	61,1	67,7	70,4	72,2	85,1	105,2
8,74	8,77	8,81	8,9	8,98	9,11	9,14	9,22	9,31	9,35

Вариант 6

3,78	3,87	4	4,23	4,33	4,59	4,87	5,14	5,59	5,61
65,72	62,05	59,7	58,1	56,8	51,8	50,8	49,7	45,8	44,5

Вариант 7

7,46	7,27	7,03	6,4	6,16	5,88	5,61	5,39	5,15	4,8
109,9	100,1	96,6	87,8	87,4	84,7	82,7	81,95	80,9	79,7

Вариант 8

4,65	4,86	4,96	5,49	5,58	5,91	6,04	6,13	6,15	6,33
92,55	91,52	90,1	79,3	75,3	70,5	68,3	62,6	58,5	57,4
		2	1			1	6	6	3

Вариант 9

4,97	4,89	4,14	3,92	3,43	3,06	2,74	2,31	1,52	1,06
30,55	28,42	27,99	25,96	24,35	22,89	22,81	15,77	14,45	8,78

Вариант 10

8,66	8,32	7,97	7,31	7,2	7,08	7,03	6,98	6,82	6,41
33,55	32,42	30,99	25,96	24,9	22,89	22,81	19,77	16,45	5,78

Вариант 11

8,6	8,31	6,97	7,31	7,0	7,01	7,5	6,9	6,03	6,01
30,55	31,42	31,99	24,96	24,2	21,1	20,91	18,9	17,0	5,28

Вариант 12

4,9	4,89	4,4	3,2	3,43	4,08	2,4	2,3	1,5	1,0
30,55	28,2	27,9	25,6	24,5	21,89	22,1	15,7	14,5	9,7

Вариант 13

7,6	7,2	7,01	6,24	6,16	5,8	4,61	5,39	5,15	4,8
109,1	100,6	96,9	87,3	87,4	83,7	81,7	81,95	80,9	79,7

Вариант 14

4,6	3,0	2,95	2,6	2,1	1,96	1,6	0,96	0,76	0,16
6,2	10,9	13,1	18,81	20,9	24,1	32,2	33,	36,6	40,45

Вариант 15

3,36	4,4	2,5	2,6	2,2	1,6	2,06	0,96	0,86	0,16
8,23	10,9	12,9	17,31	22,9	24,1	31,2	33,0	37,6	47,45

ГЛАВА 6

ЧИСЛЕННОЕ ИНТЕГРИРОВАНИЕ

Классификация методов

Ставится задача вычислить определенный интеграл вида

$$I = \int_a^b f(x) dx, \quad (6.1)$$

где $f(x)$ – непрерывная функция на интервале $[a, b]$.

Во многих случаях, когда подынтегральная функция задана в аналитическом виде, определенный интеграл удастся вычислить непосредственно с помощью неопределенного интеграла по формуле Ньютона-Лейбница. Она состоит в том, что определенный интеграл равен приращению первообразной $F(x)$ на отрезке интегрирования:

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

Однако на практике этой формулой часто нельзя воспользоваться по двум основным причинам:

- 1) вид функции $f(x)$ не позволяет аналитически выразить первообразную $F(x)$ через элементарные функции;
- 2) подынтегральная функция $f(x)$ задана таблично.

В этих случаях используют приближенные, методы интегрирования.

Сущность большинства приближенных методов вычисления определенных интегралов состоит в замене подынтегральной функции $f(x)$ аппроксимирующей функцией $\varphi(x)$, для которой можно легко записать первообразную в элементарных функциях, т.е.

$$\int_a^b f(x)dx = \int_a^b \varphi(x)dx + R, \quad (6.2)$$

где R – погрешность вычисления интеграла.

С целью уменьшения погрешности, связанной с аппроксимацией подынтегральной функции, интервал интегрирования $[a, b]$ разбивают на n отрезков и на каждом частичном отрезке заменяют подынтегральную функцию аппроксимирующей функцией $\varphi_i(x)$. Тогда приближенное значение интеграла определяется суммой частичных интегралов от функций $\varphi_i(x)$, взятых в пределах от x_{i-1} до x_i :

$$\int_a^b f(x)dx \approx \sum_{i=1}^n \int_{x_{i-1}}^{x_i} \varphi_i(x)dx. \quad (6.3)$$

Используемые на практике методы численного интегрирования можно сгруппировать в зависимости от способа аппроксимации подынтегральной функции.

Методы Ньютона-Котеса основаны на полиномиальной аппроксимации (интерполяции) подынтегральной функции. Методы этого класса отличаются друг от друга степенью используемого полинома, от которого зависит количество узлов, где необходимо вычислять функцию $f(x)$. В методах Ньютона-Котеса отрезок интегрирования разбивается, как правило, на отрезки равной длины, величина которых определяется как $h = (b - a) / n$ и называется *шагом интегрирования*. Алгоритмы этих методов просты и легко поддаются программной реализации.

Сплайновые методы базируются на аппроксимации подынтегральной функции сплайнами. Методы различаются по типу выбранных сплайнов. Такие методы имеет смысл использовать в задачах, где алгоритмы сплайновой аппроксимации применяются для обработки данных.

В *методах наивысшей алгебраической точности* используют неравноотстоящие узлы, расположенные так, чтобы обеспечить минимальную погрешность интегрирования при заданном количестве узлов. Методы различаются способом выбора узлов. Наиболее широкое применение получили *методы Гаусса*, в которых

узлы интегрирования выбираются как корни полиномов Лежандра.

В *методах Монте-Карло* узлы выбираются с помощью генератора случайных чисел, результат в итоге носит вероятностный характер. Методы оказываются особенно эффективны при вычислении кратных интегралов.

Независимо от выбранного метода в процессе численного интегрирования необходимо вычислить приближенное значение интеграла и оценить погрешность R .

Погрешность, возникающая при численном интегрировании, также как и при численном дифференцировании, имеет два основных источника. Первым источником погрешности является приближенная замена подынтегральной функции аппроксимирующей функцией – погрешность аппроксимации.

Как будет показано ниже, погрешность аппроксимации уменьшается с увеличением количества отрезков разбиения интервала интегрирования n за счет более точной аппроксимации подынтегральной функции. Другой источник погрешности – неточности в вычислении подынтегральной функции в узловых точках и ошибки округления. Эта погрешность возрастает с ростом n и с некоторого значения n^* начинает преобладать над погрешностью аппроксимации.

Это обстоятельство должно предостеречь от выбора чрезмерно большого числа n .

6.1 Интерполяционные методы Ньютона-Котеса

Способ получения формул для вычисления приближенного значения интеграла в методах Ньютона-Котеса состоит в следующем.

Разобьем интервал интегрирования $[a, b]$ на n элементарных отрезков. Точки разбиения $a = x_0, x_1, \dots, x_n = b$ будем называть *узлами интегрирования*, а расстояния между узлами $h_i = x_i - x_{i-1}$ – *шагами интегрирования*. В частном случае шаг интегрирования может быть постоянным $h = (b - a) / n$.

6.1.1 Методы прямоугольников

Рассмотрим сначала простейшие методы из класса методов Ньютона-Котеса, в которых подынтегральную функцию $f(x)$ на отрезке интегрирования $[x_{i-1}, x_i]$ заменяют полиномом нулевой степени, т.е. константой: $f(x) \approx \varphi_i(x) = c_i$.

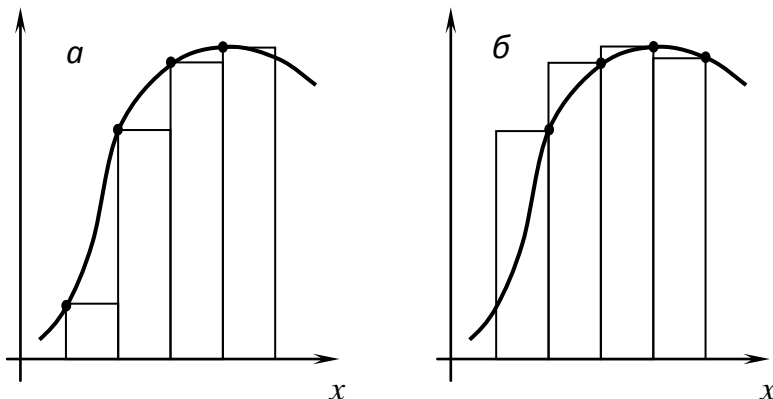


Рис. 39. Метод левых и правых прямоугольников

Подставляя $\varphi_i(x)$ в формулу (6.3) и выполняя интегрирование, получаем

$$\int_a^b f(x) dx \approx \sum_{i=1}^n c_i \int_{x_{i-1}}^{x_i} dx = \sum_{i=1}^n c_i x \Big|_{x_{i-1}}^{x_i} = \sum_{i=1}^n c_i h_i. \quad (6.4)$$

Таким образом, приближенное значение интеграла определяется суммой площадей прямоугольников, одна из сторон которых есть длина отрезка интегрирования h_i , а другая – аппроксимирующая константа. Отсюда происходит и название методов.

➤ **З а м е ч а н и е.** К формуле (6.4) можно также прийти, если заменить определенный интеграл интегральной суммой, непо-

средственно опираясь на определение понятия определенного интеграла.

Заметим, что замена подинтегральной функции константой неоднозначна, так как константу c_i можно выбрать равной значению подинтегральной функции в любой точке интервала интегрирования.

Выбирая константу c_i равную значению подинтегральной функции в левой (а) или правой (б) границах отрезка $[x_{i-1}, x_i]$ приходим к формулам *левых* и *правых прямоугольников*, соответственно:

$$\int_a^b f(x)dx \approx \sum_{i=1}^n h_i f_{i-1} = h_1 f_0 + h_2 f_1 + \dots + h_n f_{n-1}, \quad (6.5)$$

$$\int_a^b f(x)dx \approx \sum_{i=1}^n h_i f_i = h_1 f_1 + h_2 f_2 + \dots + h_n f_n. \quad (6.6)$$

В формулах (6.5) и (6.6) введены обозначения $f_{i-1} = f(x_{i-1})$, $f_i = f(x_i)$, которыми мы будем пользоваться и в дальнейшем.

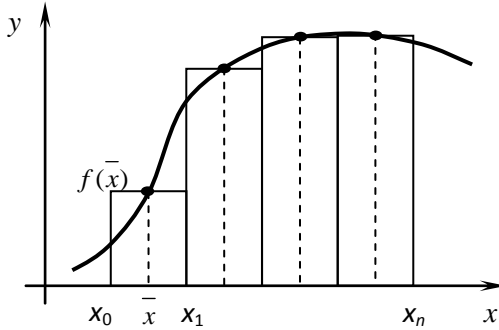
В случае постоянного шага интегрирования формулы (6.5) и (6.6) приобретают вид

$$\int_a^b f(x)dx \approx h \sum_{i=1}^n f_{i-1} = h(f_0 + f_1 + \dots + f_{n-1}), \quad (6.7)$$

$$\int_a^b f(x)dx \approx h \sum_{i=1}^n f_i = h(f_1 + f_2 + \dots + f_n). \quad (6.8)$$

Наиболее широко на практике используется формула *средних прямоугольников*, в которой значение константы c_i (высота прямоугольника) выбирается равной значению подинтегральной функции в средней точке \bar{x}_i отрезка интегрирования (рис. 5.3):

$$\int_a^b f(x)dx \approx \sum_{i=1}^n h_i f(\bar{x}_i), \quad \bar{x}_i = x_{i-1} + h_i / 2. \quad (6.9)$$



. Рис.40

В случае постоянного шага интегрирования, когда $h_i = h$, формула средних прямоугольников (5.16) будет иметь следующий вид

$$\int_a^b f(x) dx \approx h \sum_{i=1}^n f(\bar{x}_i), \quad \text{где } \bar{x}_i = a + h(i-1/2).$$

Из трех рассмотренных методов прямоугольников метод средних прямоугольников является наиболее точным.

➤ **З а м е ч а н и е.** Если подынтегральная функция $f(x)$ задана не аналитическим выражением, а таблично, то формула средних прямоугольников оказывается неприменима (без привлечения дополнительной интерполяции), так как значения функции известны лишь в узловых точках. В этом случае пользуются либо формулами левых или правых прямоугольников, либо используют другие методы.

6.1.2 Метод трапеций

На частичном отрезке интегрирования $[x_{i-1}, x_i]$ заменим подынтегральную функцию $f(x)$ интерполяционным полиномом первой степени, т.е. прямой, проходящей через точки (x_{i-1}, f_{i-1}) и (x_i, f_i) (рис. 35):

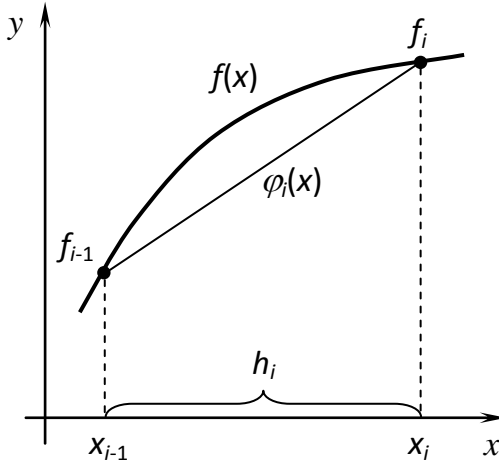


Рис. 41. Метод трапеций.

$$f(x) \approx \varphi_i(x) = f_{i-1} + \frac{f_i - f_{i-1}}{x_i - x_{i-1}}(x - x_{i-1}).$$

Поставляя это выражение в формулу (5.12) и выполняя интегрирование по частичным отрезкам, приходим к формуле трапеций:

$$\int_a^b f(x) dx \approx \frac{1}{2} \sum_{i=1}^n h_i (f_{i-1} + f_i). \quad (6.10)$$

Название метода связано с тем, что интеграл по отрезку $[x_{i-1}, x_i]$ заменяется площадью трапеции с основаниями, равными значениям функции $f(x)$ на краях отрезка, и высотой, равной h_i (рис. 35).

В случае постоянного шага интегрирования формула трапеций принимает вид

$$\int_a^b f(x) dx \approx \frac{h}{2} \sum_{i=1}^n (f_{i-1} + f_i) = \frac{h}{2} [f_0 + 2f_1 + 2f_2 + \dots + 2f_{n-1} + f_n]. \quad (6.11)$$

Для удобства вычислений и оптимизации вычислительных погрешностей формулу (6.11) часто записывают следующим образом:

$$\int_a^b f(x)dx \approx h \left(\frac{f_0 + f_n}{2} + \sum_{i=1}^{n-1} f_i \right). \quad (6.12)$$

➤ **З а м е ч а н и е.** Несмотря на то, что в методе трапеций используется более высокий порядок интерполяции подынтегральной функции (кусочно-линейная интерполяция), по сравнению с методом прямоугольников, который использует интерполяцию нулевого порядка (кусочно-постоянная), точность метода трапеций оказывается ниже точности метода средних прямоугольников. Более высокая точность метода средних прямоугольников объясняется “удачным” выбором узловых точек, в которых вычисляется высота элементарного прямоугольника.

6.1.3 Метод Симпсона (метод парабол)

Разобьем интервал интегрирования $[a, b]$ на четное число n равных отрезков с шагами h . На каждом отрезке $[x_{i-1}, x_{i+1}]$, содержащем три узла, заменим подынтегральную функцию интерполяционным полиномом второй степени, т.е. параболой

$$f(x) \approx \varphi_i(x) = f_i + \frac{f_{i+1} - f_{i-1}}{2h}(x - x_i) + \frac{f_{i-1} - 2f_i + f_{i+1}}{2h^2}(x - x_i)^2$$

Интегрируя это выражение на отрезке $[x_{i-1}, x_{i+1}]$, получим

$$\begin{aligned} \int_{x_{i-1}}^{x_{i+1}} \varphi_i(x) dx &= \int_{x_{i-1}}^{x_{i+1}} f_i + \frac{f_{i+1} - f_{i-1}}{2h}(x - x_i) + \frac{f_{i-1} - 2f_i + f_{i+1}}{2h^2}(x - x_i)^2 dx = \\ &= \frac{h}{3}(f_{i-1} + 4f_i + f_{i+1}). \end{aligned} \quad (6.13)$$

Приближенное значение интеграла на интервале $[a, b]$ получим суммированием частичных интегралов (5.21) по всем отрезкам $[x_{i-1}, x_{i+1}]$:

$$\begin{aligned} \int_a^b f(x) dx &\approx \frac{h}{3} \sum_{i=1}^{n/2} (f_{2i-2} + 4f_{2i-1} + f_{2i}) = \\ &= \frac{h}{3} (f_0 + 4f_1 + 2f_2 + 4f_3 + 2f_4 + \dots + 2f_{n-2} + 4f_{n-1} + f_n). \end{aligned} \quad (6.14)$$

Полученное соотношение называется *формулой Симпсона* или *формулой парабол*.

Если подынтегральную функцию $f(x)$ интерполировать полиномом второй степени на отрезке $[x_{i-1}, x_i]$ с привлечением дополнительной точки $x_{i-\frac{1}{2}}$ – середины отрезка, то в этом случае число отрезков разбиения n может быть произвольным (не обязательно четным), а формула Симпсона в этом случае будет иметь вид

$$\int_a^b f(x)dx \approx \frac{h}{6} \sum_{i=1}^n (f_{i-1} + 4f_{i-\frac{1}{2}} + f_i) = \frac{h}{6} (f_0 + 4f_{\frac{1}{2}} + 2f_1 + 4f_{\frac{3}{2}} + \dots + 2f_{n-1} + 4f_{n-\frac{1}{2}} + f_n). \quad (6.15)$$

Напомним, что $f_i = f(x_i)$, $f_{i-\frac{1}{2}} = f(x_i - h/2)$.

➤ **З а м е ч а н и е .** Формула (6.14) пригодна для вычисления интегралов от функций, заданных как аналитическим выражением, так и таблично, тогда как формула (6.15) применима только в тех случаях, когда подынтегральная функция задана аналитически.

6.1.4 Погрешность формул Ньютона-Котеса

Приведем здесь лишь результаты оценки погрешностей:

- методы левых и правый прямоугольников – $|R| \leq \frac{1}{2}(b-a)hM_1$;
- методы средних прямоугольников – $|R| \leq \frac{1}{24}(b-a)h^2M_2$;
- метод трапеций – $|R| \leq \frac{1}{12}(b-a)h^2M_2$;
- метод Симпсона – $|R| \leq \frac{1}{180}(b-a)h^4M_4$;

- метод Симпсона – $|R| \leq \frac{1}{2880} (b-a)h^4 M_4$.

Методы левых и правых прямоугольников являются методами первого порядка.

Методы средних прямоугольников и трапеций имеют второй порядок точности, при этом метод трапеций обладает вдвое большей по абсолютной величине погрешностью по сравнению с методом средних прямоугольников.

Поэтому, если подынтегральная функция задана аналитически, то предпочтительнее из методов второго порядка применять метод средних прямоугольников вследствие его меньшей погрешности. Метод Симпсона имеет четвертый порядок точности с очень малым численным коэффициентом.

Формула Симпсона позволяет получить очень высокую точность, если четвертая производная подынтегральной функции не слишком велика. В противном случае, методы второго порядка точности могут дать большую точность, чем метод Симпсона.

Например, для функции $f(x) = -25x^4 + 45x^2 - 7$ формула трапеций при $n = 2$ для интеграла в пределах $[-1, 1]$ дает точный результат, равный 4, тогда как по формуле Симпсона получим результат, несовпадающий даже по знаку $(-8/3)$.

6.1.5 Вычисление интеграла с заданной точностью

Используя приведенные выше оценки можно априори (до проведения расчета) определить шаг интегрирования h , при котором погрешность вычисленного результата гарантированно не превысит допустимый уровень погрешности ε . Однако на практике пользоваться априорными оценками погрешности не всегда удобно. Тогда контроль за точностью получаемого результата можно организовать следующим образом. Пусть вычисления проводились с постоянным шагом h , $I^{(h)}$ – вычисленное с шагом h приближенное значение интеграла I . Если затем вычислить приближенное значение $I^{(h/2)}$ с шагом $h/2$, то в качестве оценки погрешности последнего вычисленного значения можно рассматривать величину

$$|I^{(h/2)} - I| \approx |I^{(h/2)} - I^{(h)}|.$$

При необходимости вычислить результат с заданной точностью ε вычисления повторяют с последовательно уменьшающимся (вдвое) шагом до тех пор, пока не выполнится условие

$$|I^{(h/2)} - I^{(h)}| \leq \varepsilon. \quad (6.16)$$

Вопросы для самоконтроля для самоконтроля

1. В чем состоит идея методов Ньютона-Котеса для приближенного вычисления определенных интегралов?
2. Как влияет на точность численного интегрирования величина шага h ?
3. Можно ли добиться неограниченного уменьшения погрешности интегрирования путем последовательного уменьшения шага?
4. Почему метод средних прямоугольников неприменим для численного интегрирования таблично заданных функций?
5. Каковы преимущества формулы Симпсона по сравнению с формулой трапеций и следствием чего являются эти преимущества?
6. В каком случае формула Симпсона дает точное значение интеграла?
7. Какой подход используется на практике для вычисления интеграла с заданной точностью?

Лабораторная работа №6

Приближенное вычисление интегралов

Цель: Научиться приближенно вычислять интегралы различными методами

Задание 1. Составить в MathCAD программу вычисления интеграла от заданной функции на отрезке $[a, b]$ по формуле трапеций с точностью $\varepsilon = 10^{-4}$. Произвести оценку погрешности.

Пример выполнения задания представлен на рис. 36.

По формуле трапеций с точностью до 10^{-4} вычислить

$$\int_{1.2}^{2.6} \frac{dx}{0.6 + x^3}$$

при $n = 10, n = 20, n = 40, n = 80$, провести оценку погрешностей

$$\begin{aligned} a &:= 1.2 & b &:= 2.6 & \varepsilon &:= 10^{-4} \\ n &:= 10 & h &:= \frac{(b-a)}{n} & h &= 0.14 \\ i &:= 0..n \end{aligned}$$

$$\begin{aligned} x_0 &:= a & x_i &:= x_0 + i \cdot h & y &:= \frac{1}{0.6 + x^3} \\ I1 &:= h \cdot \left(\sum_{i=1}^{n-1} y_i + \frac{y_0 + y_n}{2} \right) & I1 &= 0.2356 \end{aligned}$$

$$\begin{aligned} n &:= 20 & h &:= \frac{(b-a)}{n} & h &= 0.07 \\ i &:= 0..n \\ x_0 &:= a & x_i &:= x_0 + i \cdot h & y &:= \frac{1}{0.6 + x^3} \\ I2 &:= h \cdot \left(\sum_{i=1}^{n-1} y_i + \frac{y_0 + y_n}{2} \right) & I2 &= 0.2347 \end{aligned}$$

$$\begin{aligned} |I2 - I1| &= 8.999 \times 10^{-4} & |I^{(h/2)} - I^{(h)}| &\leq \varepsilon \\ n &:= 40 \end{aligned}$$

$$\begin{aligned} i &:= 0..n & h &:= \frac{(b-a)}{n} & h &= 0.035 \\ x_0 &:= a & x_i &:= x_0 + i \cdot h \\ I3 &:= h \cdot \left(\sum_{i=1}^{n-1} y_i + \frac{y_0 + y_n}{2} \right) & y &:= \frac{1}{0.6 + x^3} & I3 &= 0.2345 \end{aligned}$$

$$|I3 - I2| = 2.252 \times 10^{-4}$$

$$\begin{aligned} i &:= 0..n & h &:= \frac{(b-a)}{n} & h &= 0.018 \\ x_0 &:= a & x_i &:= x_0 + i \cdot h & y &:= \frac{1}{0.6 + x^3} \end{aligned}$$

$$I4 := h \cdot \left(\sum_{i=1}^{n-1} y_i + \frac{y_0 + y_n}{2} \right) \quad I4 = 0.2344$$

$|I4 - I3| = 5.632 \times 10^{-5}$ Требуемая точность достигнута лишь при $n = 80$

Рис. 42

Варианты к заданию 1.

$$1. \int_{1,2}^{2,6} \frac{dx}{\sqrt{x^2+0,6}} \cdot 2. \int_{1,4}^{2,2} \frac{dx}{\sqrt{3x^2+1}} \cdot 3. \int_{0,8}^{1,8} \frac{dx}{\sqrt{x^2+6}}.$$

$$4. \int_{1,6}^{2,2} \frac{dx}{\sqrt{x^2+2,5}} \cdot 5. \int_{0,6}^{1,6} \frac{dx}{\sqrt{x^2+0,8}} \cdot 6. \int_{1,2}^2 \frac{dx}{\sqrt{x^2+1,2}}.$$

$$7. \int_{1,2}^{2,6} \frac{dx}{\sqrt{x^2+0,6}} \cdot 8. \int_{1,4}^2 \frac{dx}{\sqrt{2x^2+0,6}} \cdot 9. \int_{3,6}^{4,6} \frac{dx}{\sqrt{0,5x^2+1}}.$$

$$10. \int_{2,1}^{3,6} \frac{dx}{\sqrt{x^2-4}} \cdot 11. \int_{2,1}^{3,6} \frac{dx}{\sqrt{x^3-4}} \cdot 12. \int_{1,1}^{3,6} \frac{dx}{\sqrt{x^3-7}} \cdot 13.$$

$$14. \int_{1,2}^{23,6} \frac{dx}{\sqrt{x^3-5}} \cdot 15. \int_{1,1}^{1,6} \frac{dx}{\sqrt{x^2+11}}$$

Задание 2. Вычислить интеграл по формулам Симпсона и трапеций. Оценить погрешность.

N	Функция	a	b
1	$0.37e^{\sin x}$	0	1
2	$0.5x + x \ln x$	1	2
3	$(x+1.9)\sin(x/3)$	1	2
4	$\frac{1}{x} \ln(x+2)$	2	3

5	$\frac{3 \cos x}{2x+1.7}$	0	1
6	$(2x+0.6)\cos(x/2)$	1	2
7	$2.6x^2 \ln x$	1.2	2.2
8	$(x^2+1)\sin(x-0.5)$	1	2
9	$x^2 \cos(x/4)$	2	3
10	$\frac{\sin(0.2x-3)}{x^2+1}$	3	4
11	$\frac{3tgx}{2x+1.7}$	$\frac{\pi}{3}$	$\frac{\pi}{4}$
12	$\frac{7 \cos x}{2x+1.9}$	$\frac{\pi}{12}$	$\frac{\pi}{3}$
13	$\frac{\sin x}{2x-1.2}$	$\frac{\pi}{2}$	π
14	$\frac{3 \arccos x}{x-1.2}$	-0.5	1
15	$\frac{1}{x} \ln(x^2+2)$	-1.2	5

ГЛАВА 7

ЧИСЛЕННЫЕ МЕТОДЫ РЕШЕНИЯ ОБЫКНОВЕННЫХ ДИФФЕРЕНЦИАЛЬНЫХ УРАВНЕНИЙ

В настоящее время разработано большое количество методов решения дифференциальных уравнений через элементарные функции, однако на практике эти методы бывают или совсем невозможны, или слишком громоздки. Поэтому для решения практических задач созданы методы приближенного решения дифференциальных уравнений. Условно эти методы делят на три группы в зависимости от формы представления решения:

— *аналитические методы*, дающие приближенное решение в виде аналитического выражения;

— *графические методы*, дающие приближенное решение в виде графика;

— *численные методы*, дающие приближенное решение в виде таблицы.

Ниже рассматриваются следующие методы решения обыкновенных дифференциальных уравнений: интегрирование дифференциальных уравнений с помощью формулы Тейлора, метод последовательных приближений (метод Пикара), метод Эйлера и его модификации, метод Рунге-Кутты.

7.1 Интегрирование дифференциальных уравнений с помощью степенных рядов

В некоторых случаях, когда интегрирование дифференциального уравнения точными методами невозможно, его решение ищут в виде степенного ряда:

$$y(x) = \sum_{n=0}^{\infty} C_n \cdot (x - x_0)^n.$$

Неопределённые коэффициенты C_n находят подстановкой ряда в исходное уравнение, приравнивая коэффициенты при одинаковых степенях разности $x - x_0$ в обеих частях полученного равен-

ства. Если удаётся найти все коэффициенты ряда, то полученный ряд определяет решение во всей своей области сходимости.

Рассмотрим подробнее суть метода на примере решения обыкновенного дифференциального уравнения первого порядка.

Пусть задано обыкновенное, разрешённое относительно производной, дифференциальное уравнение первого порядка

$$y' = f(x, y) \quad (7.1)$$

с начальным условием

$$y(x_0) = y_0. \quad (7.2)$$

Пусть правая часть уравнения (7.1) является в точке (x_0, y_0) аналитической функцией, то есть может быть разложена в степенной ряд в некоторой окрестности этой точки. Тогда существует единственное решение $y = y(x)$ заданного уравнения (7.1) с начальным условием (7.2). Это решение также будет являться аналитическим в точке x_0 и может быть представлено в виде ряда Тейлора:

$$y(x) = \sum_{k=0}^{\infty} \frac{y^{(k)}(x_0)}{k!} \cdot (x - x_0)^k, \quad (7.3)$$

где $|x - x_0| < h$, h — некоторое положительное число; k — целое неотрицательное число.

Значения производных $y^{(k)}(x_0)$ в разложении (7.3) находят следующим образом: последовательно дифференцируют по x уравнение (7.1) $y' = f(x, y)$ $y'' = f'_x + f'_y \cdot y'$ (7.4)

$$y''' = f''_{xx} + 2f''_{xy} \cdot y' + f''_{yy} \cdot y'^2 + f'_y \cdot y'',$$

подставляют начальное условие (7.2) в формулы (7.4):

$$y'(x_0), y''(x_0), y'''(x_0), \dots$$

На практике ряд (7.3) обрывают на каком-либо члене и ищут приближённое решение в виде многочлена некоторой степени.

Аналогично с помощью ряда Тейлора можно интегрировать и дифференциальные уравнения высших порядков.

Пример.

1. Найти приближённое решение дифференциального уравнения $y' = x - 2y$, удовлетворяющее начальному условию $y(0) = 0$, в виде многочлена пятой степени.

2. Найти численное решение (в виде таблицы) данного дифференциального уравнения на $[0, 1]$ с шагом $h = 0,1$, округляя результат до 0,001.

3. Найти точное решение и сравнить его с приближённым.

Решение.

1. Будем искать приближённое решение данного дифференциального уравнения в виде (7.3):

$$y = y_0 + \frac{y_0'}{1!}(x - x_0) + \frac{y_0''}{2!}(x - x_0)^2 + \frac{y_0'''}{3!}(x - x_0)^3 + \dots$$

Так как $y_0 = 0$, тогда:

$$y_0' = 0;$$

$$y_0'' = 1 + (-2)y' \Big|_{x=0} = 1;$$

$$y_0''' = -2y'' \Big|_{x=0} = -2;$$

$$y_0^{IV} = -2y''' \Big|_{x=0} = 4;$$

$$y_0^V = -2y^{IV} \Big|_{x=0} = -8.$$

Подставив найденные значения производных в уравнение (7.3), получим требуемое приближённое решение в виде многочлена пятой степени:

$$y = \frac{x^2}{2!} - \frac{2x^3}{3!} + \frac{4x^4}{4!} - \frac{8x^5}{5!} = \frac{x^2}{2} - \frac{x^3}{3} + \frac{x^4}{6} - \frac{x^5}{15}. \quad (7.5)$$

2. Найдём приближённое решение данного дифференциального уравнения в виде таблицы.

Подставив в решение (7.5) значения аргумента x из промежутка $[0, 1]$ с шагом $h = 0,1$, получим численное решение данного уравнения на указанном промежутке. Запишем его, округляя результат до 0,001, в виде таблицы.

x	0	0,1	0,2	0,3	0,4	0,5	0,6	0,7	0,8	0,9	1
y	0	0	0,018	0,037	0,062	0,092	0,124	0,159	0,196	0,232	0,267

3. Найдём точное решение и сравним его с приближённым. Данное уравнение $y' = x - 2y$, или $y' + 2y - x = 0$, является линей-

ным дифференциальным уравнением первого порядка. Решим его методом Бернулли.

Пусть $y = u v$, тогда имеем $u'v + uv' + 2uv - x = 0$. Сгруппируем члены, содержащие u в первой степени, получим

$$u(v' + 2v) + u'v - x = 0. \text{ полагаем } v' + 2v = 0, \text{ откуда } \frac{dv}{v} = -2dx.$$

Интегрируем:

$$\int \frac{dv}{v} = -2 \int dx, \text{ откуда находим } \ln v = -2x \Rightarrow v(x) = e^{-2x}.$$

Постоянную интегрирования не вводим, так как достаточно найти какое-либо частное решение этого вспомогательного уравнения.

Для нахождения u имеем уравнение $u'e^{-2x} = x$, или $u' = x \cdot e^{2x}$. Разделяем переменные:

$$du = x \cdot e^{2x} dx,$$

интегрируем по частям:

$$\int du = \int x \cdot e^{2x} dx = \left| \begin{array}{l} \int u_1 dv_1 = u_1 v_1 - \int v_1 du_1; \\ u_1 = x; \quad dv_1 = e^{2x} dx; \\ du_1 = dx; \quad v_1 = \frac{1}{2} e^{2x} \end{array} \right| = \frac{1}{2} e^{2x} x - \frac{1}{2} \int e^{2x} dx =$$

$$\frac{1}{2} x \cdot e^{2x} - \frac{1}{4} e^{2x} + C.$$

Таким образом находим функцию $u(x)$:

$$u = \frac{1}{2} x \cdot e^{2x} - \frac{1}{4} e^{2x} + C.$$

Подставив найденные значения функций $u = u(x)$ и $v = v(x)$ в уравнение $y = u v$, получим общее решение данного уравнения:

$$y = u v = \left(\frac{1}{2} x \cdot e^{2x} - \frac{1}{4} e^{2x} + C \right) \cdot e^{-2x} = \frac{1}{2} x - \frac{1}{4} + e^{-2x} C.$$

Используя начальное условие $y(0) = 0$, получаем $0 = -\frac{1}{4} + C$,

откуда $C = \frac{1}{4}$. Следовательно, искомое частное (точное) решение

имеет вид

$$y = \frac{1}{2}x - \frac{1}{4} + \frac{1}{4}e^{-2x}. \quad (7.6)$$

Сравнение приближённого решения (7.5) данного дифференциального уравнения с точным (7.6) на промежутке $[0, 1]$

проведём с помощью компьютера

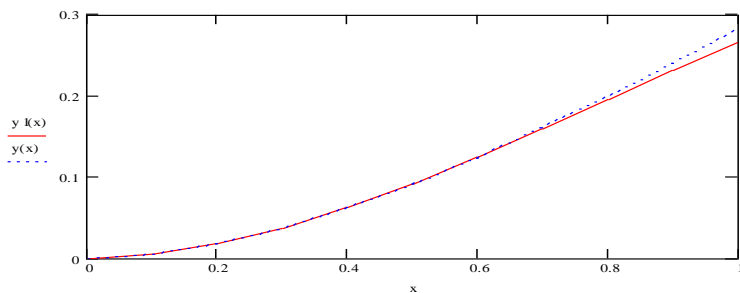
Результаты сравнения представлены на Рис. 37.

Сравнение приближённого и точного решений
дифференциального уравнения $y' = x - 2y$ на $[0, 1]$
в системе MathCad

$x := 0, 0.1, 1$ - задание промежутка изменения аргумента;

$y1(x) := \left(\frac{x^2}{2}\right) - \left(\frac{x^3}{3}\right) + \left(\frac{x^4}{6}\right) - \left(\frac{x^5}{15}\right)$ - приближённое решение диф.
уравнения в виде многочлена пятой
степени;

$y(x) := \left(\frac{x}{2}\right) - \left(\frac{1}{4}\right) + e^{-2x} \cdot \frac{1}{4}$ - точное решение диф. уравнения;



- графики приближённого $y1(x)$ и точного $y(x)$ решений уравнения

Рис. 43

7.2. Метод последовательных приближений

Метод последовательных приближений (или *метод Пикара*) является аналитическим, т. е. позволяет получить приближённое решение задачи Коши, определяемой дифференциальным урав-

нением (7.1) с начальным условием (7.2) в виде формулы. Возник метод в связи с доказательством теоремы существования и единственности решения задачи Коши.

Пусть в условиях данной теоремы требуется найти решение уравнения (7.1) с начальным условием (7.2). Проинтегрируем обе части уравнения (7.1) от x_0 до x :

$$\int_{y_0}^y dy = \int_{x_0}^x f(x, y) dx, \text{ откуда } y(x) = y_0 + \int_{x_0}^x f(x, y) dx. \quad (7.7)$$

Очевидно, что решение интегрального уравнения (7.7) будет удовлетворять уравнению (7.1) и начальному условию (7.2). Действительно, при $x = x_0$ получим

$$y(x_0) = y_0 + \int_{x_0}^{x_0} f(x, y) dx = y_0.$$

Применим к интегральному уравнению (7.7) метод последовательных приближений. Заменим в равенстве неизвестную функцию у данным значением y_0 , получим первое приближение

$$y_1(x) = y_0 + \int_{x_0}^x f(x, y_0) dx.$$

Заметим, что интеграл в правой части содержит только одну переменную x , поэтому аналитическое выражение первого приближения $y_1(x)$ будет являться функцией, зависящей от x . Заменим теперь в равенстве (7.7) неизвестную функцию y найденным значением $y_1(x)$, получим второе приближение $y_2(x) = y_0 +$

$\int_{x_0}^x f(x, y_1) dx$ и т. д. В общем случае итерационная формула имеет

вид

$$y_n(x) = y_0 + \int_{x_0}^x f(x, y_{n-1}) dx \quad (n = 1, 2, \dots). \quad (7.8)$$

Применив неоднократно формулу (7.8), получим последовательность функций

$$y_1(x), y_2(x), \dots, y_n(x), \dots. \quad (7.9)$$

Можно доказать, что эта последовательность сходится и

$$\lim_{n \rightarrow \infty} y_n(x) = y(x),$$

т.е. предел последовательности является решением интегрального уравнения (7), а следовательно, и дифференциального уравнения (1) с начальным условием (2). Это означает, что k -й член последовательности (9) является приближением к точному решению уравнения (1) с определённой степенью точности.

Погрешность k -го приближения можно оценить формулой

$$|y(x) - y_k(x)| \leq L^k \cdot M \cdot \frac{d^{k+1}}{(k+1)!}, \quad (7.10)$$

где L - постоянная Липшица; M - верхняя грань модуля функции f , т.е. $|f(x, y)| \leq M$; величина d для определения окрестности

$|x - x_0| \leq d$ вычисляется по формуле $d = \min(a, \frac{b}{M})$, числа a и

b - из неравенства Липшица.

Пример 1. Найти три последовательных приближения решения дифференциального уравнения $y' = x + y^2$, удовлетворяющего начальному условию $y(0) = 1$.

Решение.

В качестве начального приближения возьмём $y_0 = y(0) = 1$, тогда:

первое приближение $y_1(x) = y_0 + \int_{x_0}^x f(x, y_0) dx = 1 + \int_0^x (x+1) dx$,

второе приближение

$y_2(x) = y_0 + \int_{x_0}^x f(x, y_1) dx = 1 + \int_0^x (x + (y_1(x))^2) dx$, третье прибли-

жение $y_3(x) = y_0 + \int_{x_0}^x f(x, y_2) dx = 1 + \int_0^x (x + (y_2(x))^2) dx$.

Вычисления интегралов и построение графиков полученных функций $y_1(x)$, $y_2(x)$, $y_3(x)$ проведём в системе MathCAD. Результаты решения представлены на рис. 38.

Пример решения дифференциального уравнения $y' = x + y^2$,
 $y(0) = 1$ методом Пикара в системе MathCad

$x := 0, 0.1..1$ - задание промежутка изменения аргумента;

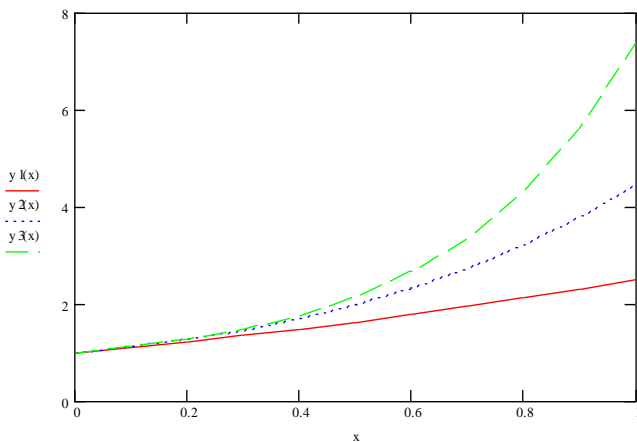
$$y1(x) := 1 + \int_0^x (1 + t) dt \rightarrow 1 + x + \frac{1}{2} \cdot x^2 \quad \text{- первое приближение;}$$

$$y2(x) := 1 + \int_0^x t + \left[1 + t + \left(\frac{1}{2} \right) \cdot t^2 \right]^2 dt \rightarrow 1 + \frac{3}{2} \cdot x^2 + \frac{1}{20} \cdot x^5 + \frac{1}{4} \cdot x^4 + \frac{2}{3} \cdot x^3 + x$$

- второе приближение;

$$y3(x) := 1 + \int_0^x t + \left[1 + \left(\frac{3}{2} \right) \cdot t^2 + \left(\frac{1}{20} \right) \cdot t^5 + \left(\frac{1}{4} \right) \cdot t^4 + \left(\frac{2}{3} \right) \cdot t^3 + t \right]^2 dt \quad \text{- третье приближение;}$$

$$1 + \frac{3}{2} \cdot x^2 + \frac{49}{60} \cdot x^5 + \frac{1}{4400} \cdot x^{11} + \frac{13}{12} \cdot x^4 + \frac{1}{400} \cdot x^{10} + \frac{4}{3} \cdot x^3 + \frac{31}{2160} \cdot x^9 + x + \frac{29}{480} \cdot x^8 + \frac{233}{1260} \cdot x^7 + \frac{13}{30} \cdot x^6$$



- графики полученных приближений

Рис. 44

7.3. Метод Эйлера

Метод Эйлера относится одновременно к численным и к графическим методам решения дифференциальных уравнений.

Суть метода заключается в том, что искомую интегральную кривую $y = y(x)$ заменяют ломаной $M_0M_1M_2 \dots$, звенья которой являются касательными к интегральной кривым (Рис.39).

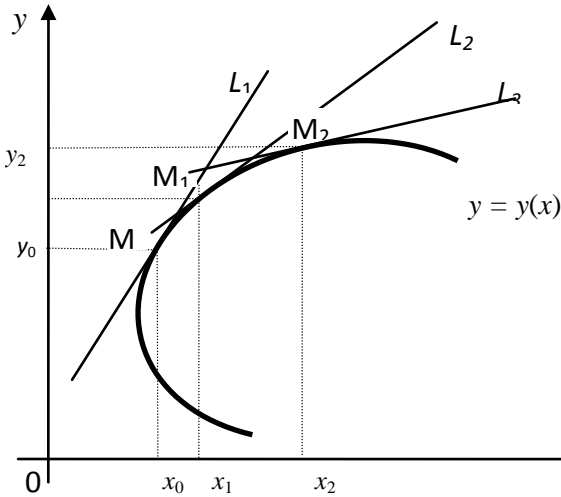


Рис. 39

Пусть требуется решить задачу Коши, т.е. найти решение дифференциального уравнения (7.1) с начальным условием (7.2) в виде функции $y = y(x)$. Выбрав шаг h , построим, начиная с точки x_0 , систему равноотстоящих точек:

$$x_i = x_0 + ih \quad (i = 0, 1, 2, \dots).$$

Вместо искомой интегральной кривой $y = y(x)$ на отрезке $[x_0, x_1]$ рассмотрим отрезок касательной L_1 к ней в точке $M_0(x_0, y_0)$.

Уравнение касательной L_1 , в силу (7.1), имеет вид

$$y = y_0 + f(x_0, y_0)(x - x_0).$$

При $x = x_1$ из уравнения касательной L_1 получим

$$y_1 = y_0 + hf(x_0, y_0),$$

откуда видим, что приращение функции на первом шаге имеет вид

$$\Delta y_0 = hf(x_0, y_0).$$

Аналогично, проводя касательную L_2 к некоторой интегральной кривой семейства в точке $M_1(x_1, y_1)$, получим

$$y = y_1 + f(x_1, y_1)(x - x_1).$$

При $x = x_2$ имеем $y_2 = y_1 + hf(x_1, y_1)$, т.е. y_2 получается из y_1 добавлением приращения

$$\Delta y_1 = hf(x_1, y_1).$$

Таким образом, значения искомой функции $y(x)$ могут быть определены по формулам:

$$y_{i+1} = y_i + \Delta y_i, \quad \Delta y_i = hf(x_i, y_i), \quad (7.11)$$

где $i = 0, 1, 2, \dots$, которые называются *вычислительными формулами метода Эйлера*.

При этом искомую интегральную кривую $y = y(x)$, проходящую через точку $M_0(x_0, y_0)$, приближённо заменяем так называемой *ломаной Эйлера* $M_0M_1M_2 \dots$, звенья которой M_iM_{i+1} прямолинейны между прямыми $x = x_i$, $x = x_{i+1}$ и имеют подъём

$$\frac{y_{i+1} - y_i}{h} = f(x_i, y_i).$$

Метод Эйлера является простейшим численным методом, удобным в применении, однако он имеет ряд существенных недостатков. Основной из них — малая точность. Она равна порядку h^2 , причём с каждым шагом погрешность возрастает, т.е. происходит систематическое накопление ошибок. Поэтому на практике часто используют способ двойного счёта — с шагом h и с шагом $h/2$. Совпадение десятичных знаков в полученных двумя способами результатах даёт естественные основания считать их верными.

Пример.

1. Найти методом Эйлера численное решение дифференциального уравнения $y' = x^3 + y$, удовлетворяющее начальному условию $y(0) = 1$, на отрезке $[0, 1]$ с шагом $h = 0,1$.

2. Найти точное решение уравнения $y' = x^3 + y$ и сравнить его с приближённым на отрезке $[0, 1]$.

Решение.

1. Для данного уравнения вычислительные формулы (7.11) имеют вид:

$$y_{i+1} = y_i + \Delta y_i,$$

$$\Delta y_i = 0,1 (x_i^3 + y_i), \quad i = 0, 1, 2, \dots, 10.$$

Учитывая, что погрешность метода имеет порядок $h^2 = 0,01$, достаточно в промежуточных результатах брать три цифры после запятой, а во всех y_i сохранять только две цифры.

Результаты вычислений оформим в виде таблицы.

i	x_i	y_i	$\Delta y_i = hf(x_i, y_i) = 0,1(x_i^3 + y_i)$
0	0	1	0,1
1	0,1	1,1	0,110
2	0,2	1,21	0,122
3	0,3	1,33	0,136
4	0,4	1,47	1,634
5	0,5	1,62	0,175
6	0,6	1,79	0,201
7	0,7	1,99	0,233
8	0,8	2,22	0,273
9	0,9	2,49	0,322
10	1	2,82	—

2. Данное уравнение $y' = x^3 + y$ является линейным дифференциальным уравнением первого порядка. Решим его методом Бернулли. Полагая $y = u v$, имеем

$$u'v + uv' - uv - x^3 = 0.$$

Сгруппируем члены, содержащие u в первой степени, получим

$$u(v' - v) + u'v - x^3 = 0. \text{ Полагаем } v' - v = 0, \text{ откуда } \frac{dv}{v} = dx.$$

Интегрируя, находим $\ln v = x$, или $v = e^x$ (постоянную интегри-

рования не вводим, так как достаточно найти какое-либо частное решение этого вспомогательного уравнения).

Для нахождения u имеем уравнение $u'e^x = x^3$, или $u' = x^3 \cdot e^{-x}$. Разделим переменные, получим $du = x^3 \cdot e^{-x} dx$, откуда

$$\int du = \int x^3 \cdot e^{-x} dx.$$

Интегрируем по частям три раза:

$$\begin{aligned} u &= -e^{-x} \cdot x^3 + 3 \int e^{-x} \cdot x^2 dx = e^{-x} \cdot x^3 + 3 \cdot (e^{-x} \cdot x^2 + 2 \int e^{-x} \cdot x dx) \\ &= -e^{-x} \cdot x^3 - 3x^2 \cdot e^{-x} + 6 \cdot (-e^{-x} \cdot x + \int e^{-x} dx) = \\ &= -e^{-x} (x^3 + 3x^2 + 6x + 6) + C. \end{aligned}$$

Таким образом, общее решение данного уравнения

$$y = u v = (-e^{-x} (x^3 + 3x^2 + 6x + 6) + C) \cdot e^x,$$

или $y = Ce^x - x^3 - 3x^2 - 6x - 6$.

Используя начальное условие $y(0) = 1$, получим $1 = -6 + C$, откуда $C = 7$. Следовательно, искомое частное (точное) решение имеет вид

$$y = 7e^x - x^3 - 3x^2 - 6x - 6.$$

Вычислим значения полученного точного решения на отрезке $[0, 1]$ с шагом $h = 0,1$. Результаты округлим до 0,01 и запишем в таблицу.

i	x_i	Приближённые значения y_i	Точные значения $y(x_i)$
0	0	1	1
1	0,1	1,1	1,11
2	0,2	1,21	1,22
3	0,3	1,33	1,35
4	0,4	1,47	1,5
5	0,5	1,62	1,67
6	0,6	1,79	1,86
7	0,7	1,99	2,08
8	0,8	2,22	2,35
9	0,9	2,49	2,66
10	1	2,82	3,03

Сравнение приближённого (численного) решения данного дифференциального уравнения с точным на промежутке $[0, 1]$ проведём с помощью системы MathCAD.

Результаты сравнения, а также численное решение данного уравнения, проведённое методом Эйлера в системе MathCAD, представлены на Рис. 40.

Пример решения дифференциального уравнения $y' = y + x^3$, $y(0)=1$ методом Эйлера в системе MathCad

$f(x, y) := y + x^3$ - задание правой части уравнения;

$x := 0, 0.1.. 1$ - задание промежутка изменения аргумента;

$i := 0.. 9$ - задание цикла; $h := 0.1$ - задание шага;

$\begin{bmatrix} x_0 \\ y_0 \end{bmatrix} := \begin{bmatrix} 0 \\ 1 \end{bmatrix}$ - задание начальных условий в векторном виде;

$\begin{bmatrix} x_{i+1} \\ y_{i+1} \end{bmatrix} := \begin{bmatrix} x_i + h \\ y_i + h \cdot f(x_i, y_i) \end{bmatrix}$ - задание вычислительных формул;

$y1(x) := 7 \cdot e^x - x^3 - 3 \cdot x^2 - 6 \cdot x - 6$ - точное решение;

Результаты вычислений: $y1(x_i)$ $y_i - y1(x_i)$

$x =$

0	0
1	0.1
2	0.2
3	0.3
4	0.4
5	0.5
6	0.6
7	0.7
8	0.8
9	0.9
10	1

$y =$

0	1
1	1.1
2	1.21
3	1.332
4	1.468
5	1.621
6	1.796
7	1.997
8	2.231
9	2.505
10	2.828

1
1.105
1.222
1.352
1.499
1.666
1.859
2.083
2.347
2.658

0
$-5.196 \cdot 10^{-3}$
-0.012
-0.02
-0.031
-0.045
-0.063
-0.087
-0.116
-0.153

- значения приближённого y и точного $y1(x)$ решений на $[0, 1]$ с шагом $h = 0.1$;

Графики точного и приближённого решений на $[0, 1]$:

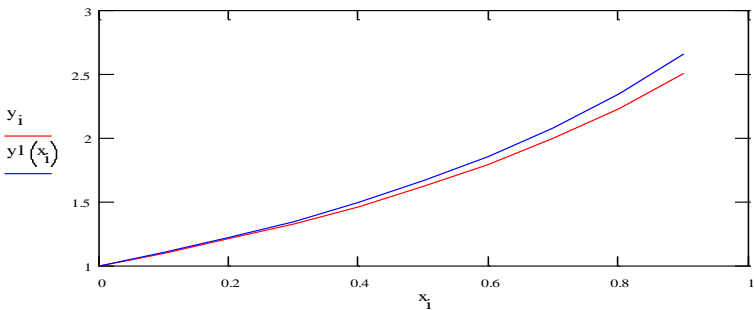


Рис. 45. Метод Эйлера

7.4. Модификации метода Эйлера

Существуют различные уточнения метода Эйлера, повышающие его точность. Цель модификаций — более точно определить направление перехода из точки (x_i, y_i) в точку (x_{i+1}, y_{i+1}) . Так, *метод Эйлера-Коши* предлагает вычислять значения искомой функции $y(x)$ по формулам:

$$y_{i+1}^* = y_i + \Delta y_i, \quad \Delta y_i = hf(x_i, y_i),$$
$$y_{i+1} = y_i + h \cdot \frac{f(x_i, y_i) + f(x_{i+1}, y_{i+1}^*)}{2}, \quad i = 0, 1, 2, \dots$$

Геометрически это означает, что мы определяем направление интегральной кривой в исходной точке (x_i, y_i) и во вспомогательной точке (x_{i+1}, y_{i+1}^*) . В качестве окончательного берём среднее этих направлений.

Другой модификацией метода Эйлера является *усовершенствованный метод ломаных*, при котором сначала вычисляют промежуточные значения:

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

и находят значение направления поля интегральных кривых в средней точке $(x_{i+\frac{1}{2}}, y_{i+\frac{1}{2}})$, т.е. $f_{i+\frac{1}{2}} = f(x_{i+\frac{1}{2}}, y_{i+\frac{1}{2}})$, а затем

полагают

$$y_{i+1} = y_i + h f_{i+\frac{1}{2}}.$$

Метод Эйлера и его модификации являются простейшими представителями *конечно-разностных* методов (*шаговых* методов) для приближённого решения задачи Коши.

Поскольку описанные методы предполагают повторяющиеся вычисления на каждом шаге, то они легко программируются и могут быть реализованы на компьютере.

На Рис. 41 и 42 показаны решения дифференциального уравнения $y' = x^3 + y$, удовлетворяющего начальному условию $y(0) = 1$,

полученные модифицированными методами Эйлера (методом Эйлера-Коши и усовершенствованным методом ломаных) с помощью системы MathCAD.

Пример решения дифференциального уравнения $y' = y + x^3$, $y(0)=1$ методом Эйлера-Коши в системе MathCad

$f(x, y) := y + x^3$ - задание правой части уравнения;

$x := 0, 0.1.. 1$ - задание промежутка изменения аргумента;

$i := 0.. 9$ - задание цикла; $h := 0.1$ - задание шага;

$\begin{bmatrix} x_0 \\ y_0 \end{bmatrix} := \begin{bmatrix} 0 \\ 1 \end{bmatrix}$ - задание начальных условий в векторном виде;

$\begin{bmatrix} x_{i+1} \\ y_{i+1} \end{bmatrix} := \begin{bmatrix} x_i + h \\ y_i + \frac{h}{2} \cdot (f(x_i, y_i) + f(x_i + h, y_i + h \cdot f(x_i, y_i))) \end{bmatrix}$ - вычислительные формулы;

Результаты вычислений:

	0		0
0	0	0	1
1	0.1	1	1.10505
2	0.2	2	1.22154
3	0.3	3	1.35159
4	0.4	4	1.49819
5	0.5	5	1.66527
6	0.6	6	1.8578
7	0.7	7	2.08189
8	0.8	8	2.34496
9	0.9	9	2.65579
10	1	10	3.02474

- численное решение $y = y(x)$ на $[0, 1]$ с шагом $h = 0.1$;

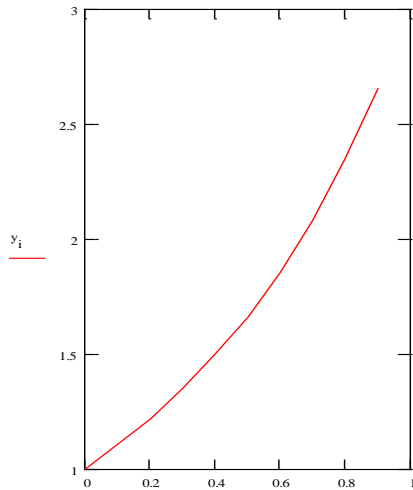


График полученного решения

Рис. 46

Пример решения дифференциального уравнения $y' = y + x^3$, $y(0)=1$ методом ломаных в системе MathCad

$f(x, y) := y + x^3$ - задание правой части уравнения;

$x := 0, 0.1.. 1$ - задание промежутка изменения аргумента;

$i := 0.. 9$ - задание цикла; $h := 0.1$ - задание шага;

$\begin{bmatrix} x_0 \\ y_0 \end{bmatrix} := \begin{bmatrix} 0 \\ 1 \end{bmatrix}$ - задание начальных условий в векторном виде;

$\begin{bmatrix} x_{i+1} \\ y_{i+1} \end{bmatrix} := \begin{bmatrix} x_i + h \\ y_i + h \cdot f\left(x_i + \frac{h}{2}, y_i + \frac{h}{2} f(x_i, y_i)\right) \end{bmatrix}$ - задание вычислительных формул;

Результаты вычислений:

	0		0
0	0	1	1.10501
1	0.1	2	1.22138
2	0.2	3	1.35123
3	0.3	4	1.49753
4	0.4	5	1.6642
5	0.5	6	1.85621
6	0.6	7	2.07965
7	0.7	8	2.34192
8	0.8	9	2.65179
9	0.9	10	3.01961
10	1		

График полученного решения

- численное решение $y = y(x)$ на $[0, 1]$ с шагом $h = 0.1$;

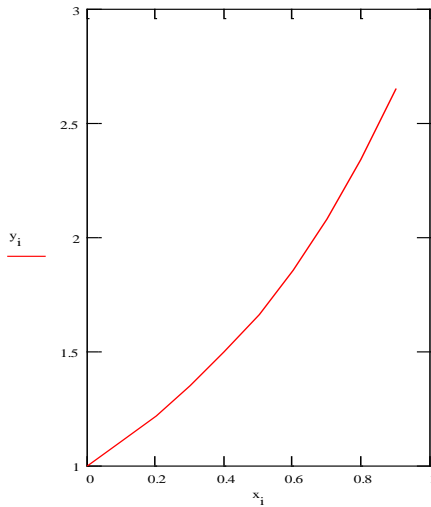


Рис. 47. Метод ломаных

7.5. Метод Рунге-Кутта

Рассмотренный выше метод Эйлера относится к семейству методов Рунге-Кутта и является их простейшим частным случаем

(методом первого порядка точности). Наиболее известным из методов Рунге-Кутты является классический четырёхэтапный метод четвёртого порядка точности. Его расчётные формулы для решения задачи Коши, определённой уравнениями (7.1) и (7.2), имеют вид:

$$y_{i+1} = y_i + \Delta y_i; \quad \Delta y_i = \frac{1}{6} (k_1^{(i)} + 2k_2^{(i)} + 2k_3^{(i)} + k_4^{(i)}), \quad (7.12)$$

где $k_1^{(i)} = h f(x_i, y_i)$;

$$k_2^{(i)} = h f\left(x_i + \frac{h}{2}, y_i + \frac{k_1^{(i)}}{2}\right);$$

$$k_3^{(i)} = h f\left(x_i + \frac{h}{2}, y_i + \frac{k_2^{(i)}}{2}\right);$$

$$k_4^{(i)} = h f(x_i + h, y_i + k_3^{(i)}), \quad i = 0, 1, 2, \dots$$

Погрешность метода на каждом шаге является величиной порядка h^5 . Геометрический смысл использования метода Рунге-Кутты с вычислительными формулами (7.12) состоит в следующем (Рис. 43).

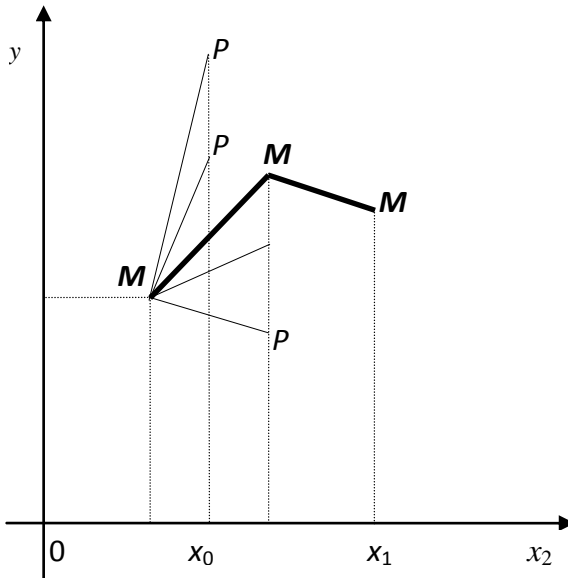


Рис. 48

Из начальной точки $M_0(x_0, y_0)$ сдвигаются в направлении, определяемом углом α_1 , для которого $\operatorname{tg} \alpha_1 = f(x_0, y_0)$. Идут в этом направлении $\frac{h}{2}$ и на полшага, т.е. до вертикальной прямой $x = x_0 + \frac{h}{2}$. На этом направлении выбирается точка P_1 с координатами

$$\left(x_0 + \frac{h}{2}; y_0 + \frac{k_1^{(0)}}{2} \right)$$

Затем из точки $M_0(x_0, y_0)$ сдвигаются в направлении, определяемом углом α_2 , для которого $\operatorname{tg} \alpha_2 = f\left(x_0 + \frac{h}{2}, y_0 + \frac{k_1^{(0)}}{2}\right)$, и на этом направлении выбирается точка P_2 с координатами

$$\left(x_0 + \frac{h}{2}; y_0 + \frac{k_2^{(0)}}{2} \right).$$

Далее из точки $M_0(x_0, y_0)$ сдвигаются в направлении, определяемом углом α_3 , для которого $\operatorname{tg} \alpha_3 = f\left(x_0 + \frac{h}{2}; y_0 + \frac{k_2^{(0)}}{2}\right)$. На

этом направлении выбирается точка P_3 с координатами

$(x_0 + h, y_0 + k_3^{(0)})$. Этим задаётся ещё одно направление, определяемое углом α_4 , для которого $\operatorname{tg} \alpha_4 = f(x_0 + h, y_0 + k_3^{(0)})$. Четыре полученных направления усредняются в соответствии с формулой

$$\Delta y_0 = \frac{1}{6}(k_1^{(0)} + 2k_2^{(0)} + 2k_3^{(0)} + k_4^{(0)}).$$

На этом окончательном направлении и выбирается очередная точка M_1 с координатами $(x_1, y_1) = (x_0 + h, y_0 + \Delta y_0)$.

Теперь, уже исходя из точки M_1 , все построения с помощью усреднений направлений повторяют сначала. Идут в новом усреднённом направлении до вертикальной прямой $x = x_2$, получают точку $M_2(x_2, y_2)$ и т.д.

Эффективная оценка метода Рунге-Кутты затруднительна [2, 4]. Поэтому для определения правильности выбора шага h на

практике обычно на каждом этапе из двух шагов применяют двойной пересчёт, а именно: исходя из текущего верного значения $y(x_i)$ вычисляют величину $y(x_i + 2h)$ двумя способами: один раз с шагом h , другой раз — с двойным шагом $2h$.

Если расхождение полученных значений не превышает допустимой погрешности, то шаг h для данного этапа выбран правильно и полученное с его помощью значение можно принять за $y(x_i + 2h)$. В противном случае шаг уменьшают в два раза.

На практике при вычислениях по формулам (7.12) обычно пользуются схемой, приведённой в таблице.

i	x	Y	$k = hf(x, y)$	Δy
0	x_0	y_0		
	$x_0 + \frac{h}{2}$	$y_0 + \frac{k_1^{(0)}}{2}$	$k_1^{(0)}$	$k_1^{(0)}$
	$x_0 + \frac{h}{2}$	$y_0 + \frac{k_2^{(0)}}{2}$	$k_2^{(0)}$	$2k_2^{(0)}$
	$x_0 + h$	$y_0 + k_3^{(0)}$	$k_3^{(0)}$	$2k_3^{(0)}$
			$k_4^{(0)}$	$k_4^{(0)}$
—	—	—	—	$\frac{1}{6} \Sigma = \Delta y_0$
1	x_1	y_1

Пример.

Найти методом Рунге-Кутты решение дифференциального уравнения $y' = x^3 + y$, удовлетворяющего начальному условию $y(0) = 1$, на отрезке $[0, 1]$ с шагом $h = 0,1$.

Решение.

Учитывая, что погрешность метода имеет порядок $h^5 = 0,00001$, в промежуточных результатах следует брать пять цифр после запятой, а во всех y_i сохранять только четыре цифры.

Результаты вычислений оформим в виде таблицы.

i	x	y	$k = 0,1(x^3 + y)$	Δy
0	0	1	0,1	0,1
	0,05	1,05	1,10501	0,21003
	0,05	1,0525	1,10526	0,21053
	0,1	1,1053	1,11063	0,11063
				0,1052
1	0,1	1,1052	0,11062	0,11062
	0,15	1,1604	0,11637	0,23278
	0,15	1,1634	0,11668	0,21121
	0,2	1,2219	0,11136	0,11136
				0,10556
2	0,2	1,2218	0,12188	0,12188
	0,25	1,2717	0,12874	0,25747
	0,25	1,2752	0,12908	0,25816
	0,3	1,3399	0,13669	0,13669
				0,12903
3	0,3	1,3520	0,13668	0,13668
	0,35	1,4081	0,1451	0,2902
	0,35	1,4124	0,14552	0,29105
	0,4	1,4853	0,15493	0,15493
				0,14548
4	0,4	1,4988	0,15493	0,15493
	0,45	1,5628	0,16539	0,33078
	0,45	1,568	0,16591	0,33182
	0,5	1,6512	0,17762	0,17762
				0,16586
5	0,5	1,6661	0,17762	0,17762
	0,55	1,74	0,19064	0,38128
	0,55	1,7465	0,19132	0,38258
	0,6	1,8425	0,20585	0,20585
				0,19122
6	0,6	1,8588	0,20584	0,20584
	0,65	1,9618	0,22199	0,44399
	0,65	1,9699	0,2228	0,4456
	0,7	2,0826	0,24082	0,24082
				0,22271
7	0,7	2,0833	0,24081	0,24081
	0,75	2,1855	0,26074	0,52148
	0,75	2,1955	0,26173	0,52347
	0,8	2,3268	0,28388	0,28388

				0,26161
8	0,8	2,3468	0,28589	0,28589
	0,85	2,4898	0,3104	0,62079
	0,85	2,5021	0,31162	0,62324
	0,9	2,6585	0,33875	0,33875
				0,31145
9	0,9	2,6582	0,34129	0,34129
	0,95	2,8545	0,37119	0,74238
	0,95	2,8695	0,37269	0,74537
	1	3,0566	0,40566	0,40566
				0,37245
10	1	3,0280		

Соответствующее решение данного дифференциального уравнения, полученное методом Рунге-Кутты в системе MathCAD, представлено на Рис. 44-45

Пример решения дифференциального уравнения $y' = x^3 + y$
 $y(0)=1$ методом Рунге-Кутты в системе MathCad

$f(x, y) := y + x^3$ - задание правой части уравнения;

$x := 0, 0.1.. 1$ - задание промежутка изменения аргумента;

$i := 0.. 9$ - задание цикла; $h := 0.1$ - задание шага;

$$k1(x, y) := h \cdot f(x, y) \quad k2(x, y) := h \cdot f\left(x + \frac{h}{2}, y + \frac{k1(x, y)}{2}\right)$$

$$k3(x, y) := h \cdot f\left(x + \frac{h}{2}, y + \frac{k2(x, y)}{2}\right)$$

$$k4(x, y) := h \cdot f(x + h, y + k3(x, y))$$

$$k(x, y) := \frac{(k1(x, y) + 2 \cdot k2(x, y) + 2 \cdot k3(x, y) + k4(x, y))}{6}$$

Рис. 49

$$\begin{pmatrix} x_0 \\ y_0 \end{pmatrix} := \begin{pmatrix} 0 \\ 1 \end{pmatrix} \quad - \text{ задание начальных условий в векторном виде;}$$

$$\begin{pmatrix} x_{i+1} \\ y_{i+1} \end{pmatrix} := \begin{pmatrix} x_i + h \\ y_i + k(x_i, y_i) \end{pmatrix} \quad - \text{ вычислительные формулы;}$$

Результаты вычислений:

	0
0	0
1	0.1
2	0.2
3	0.3
4	0.4
5	0.5
6	0.6
7	0.7
8	0.8
9	0.9
10	1

 $x =$

	0
0	1
1	1.1052
2	1.22182
3	1.35201
4	1.49877
5	1.66605
6	1.85883
7	2.08327
8	2.34678
9	2.65822
10	3.02797

 $y =$

- численное решение $y = y(x)$
на $[0, 1]$ с шагом $h = 0.1$;

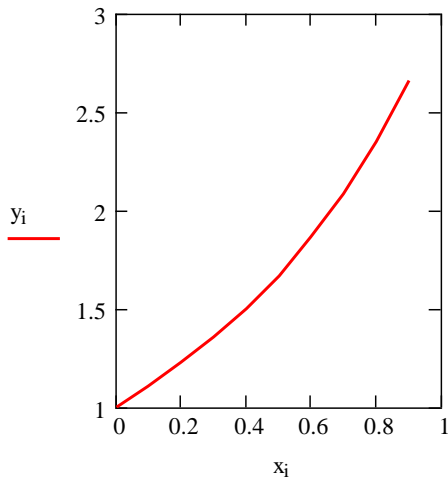


График
полученного
решения

Рис. 50. Метод Рунге-Кутта

Лабораторная работа №6

Численные методы решения задачи Коши для обыкновенных дифференциальных уравнений

Цель: научиться решать дифференциальные уравнения различными приближенными методами: методом Эйлера, методом Эйлера-Коши, методом Рунге-Кутты

Задание 1.

Решить задачу Коши для обыкновенного дифференциального уравнения первого порядка $y' = f(x, y)$ на отрезке $[a, b]$ при заданном начальном условии $y(a) = c$ и шаге интегрирования h :

- 1) методом Эйлера с шагом $2h$ и с шагом h ;
- 2) модифицированным методом Эйлера (методом Эйлера - Коши или усовершенствованным методом ломаных);
- 3) методом Рунге-Кутта с шагом $2h$ и с шагом h .

Результаты округлить до 0,0001. Сравнить полученные разными методами решения. Построить графики полученных решений.

Вариант	$f(x, y)$	a	b	c	h
1	$2 - \sin(x + y)^2$	2	3	2,3	0,1
2	$\cos(1,5x - y^2) - 1,3$	-1	1	0,2	0,2
3	$\cos(1,5y + x^2) + 1,4$	1	2	0,9	0,1
4	$\cos(0,6 + y) + 2,5x$	1	3	1,5	0,2
5	$1,5 + \sin(x + y)$	1,5	2,5	0,5	0,1
6	$\sqrt{4x^2 + 1} - 3y^2$	2,6	4,6	1,8	0,2
7	$\sqrt{x^2 + 0,5y^2} + 1$	0	2	2,9	0,2

8	$\frac{(x+y)(1-xy)}{x+2y}$	0	2	1	0,2
9	$\frac{\operatorname{tg} x - y}{\cos^2 x}$	0	2	0	0,2
10	$\arcsin x + x - \frac{xy}{1-x^2}$	0	1	3	0,1
11	$4,1x - y^2 + 0,6$	0,6	2,6	3,4	0,2
12	$e^{-(y^2+1)} + 2x$	0	0,5	0,3	0,05
13	$\frac{1}{1+x^3y} + 2y$	1,5	2	2,1	0,05
14	$\frac{2}{2+x} + x + 1$	0,1	0,5	1,25	0,05
15	$\frac{2xy}{4+x} - 0,4$	3	5	1,7	0,2

Библиографический список

1. Бахвалов, Н.С., Лапин, А.В., Чижов, Е.В. Численные методы в задачах и упражнениях. Учебное пособие. /Под ред. Садовниченко- М.: Высш. шк. 2009.- 190 с.
2. Вержбицкий, В.М. Основы численных методов: Учебник для вузов/ В.М. Вержбицкий.- 3-е изд., стер. – М.: Высш. шк., 2009. - 280 с.: ил.
3. Демидович, Б. П., Марон, И.А. Основы вычислительной математики: Учебное пособие. 7-е изд., стер. – СПб.: Издательство «Лань», 2009.- 672 с.: ил.- (Учебники для вузов. Специальная литература).
4. Калиткин, Н.Н. Численные методы: Учебное пособие. 2-е изд., испр.– СП-б.: Издательство БХВ, Петербург, 2011, 586 с.
5. Копченова, Н.В., Марон, И.А. Вычислительная математика в примерах и задачах: Учебное пособие. 2-е изд., стер. – СП-б.: Издательство Лань., 2008. -368 с.- (Учебники для вузов. Специальная литература).
6. Лапчик, М.П. Численные методы: Учебное пособие для студ. Вузов/М.П. Лапчик, М.И. Рагулина, Е.К. Хеннер; Под ред. М.П. Лапчика.- М.: Издательский центр «Академия», 2004.- 384 с.
7. Плис, А.И., Сливина, Н.А. Mathcad: математический практикум для экономистов и инженеров: Учеб. пособие.– М.: Финансы и статистика, 2003. – 656с.: ил.
8. Швецов, Г.С. Численные методы линейной алгебры: Учебное пособие/ Г.С. Швецов, О.Г. Крюкова, Б.И. Мызникова.- М.: Финансы и статистика: ИНФРА-М, 2008.- 480 с.