

Политех Петра Великого, 2 курс, осень 2022/23

Подготовка к экзамену по вычислительной математике

Лектор: Устинов Сергей Михайлович

Содержание

1. Конечные разности и их свойства. Таблица конечных разностей.	4
1.1 Конечные разности и их свойства.	4
1.2 Таблица конечных разностей.	5
2. Суммирование функций. Формула Абеля суммирования по частям.	5
2.1 Суммирование функций.	5
2.2 Суммирование по частям.	7
3. Разностное уравнение, его порядок. Линейные разностные уравнения первого порядка и порядка выше первого.	8
3.1 Разностное уравнение, его порядок.	8
3.2 Линейное разностное уравнение первого порядка	10
3.3 Линейное разностное уравнение порядка выше первого	10
4. Разделенные разности и их связь с конечными разностями.	12
5. Аппроксимация функций. Задача интерполирования.	13
5.1 Аппроксимация функций.	13
5.2 Постановка задачи интерполирования	14
6. Интерполяционный полином Лагранжа. Остаточный член полинома Лагранжа.	15
7. Выбор узлов интерполирования. Интерполяционный полином Ньютона для равно и неравноотстоящих узлов.	16
7.1 Выбор узлов интерполирования	16
7.2 Интерполяционный полином Ньютона для равно и неравноотстоящих узлов.	17
8. Сплайн-интерполяция. Подпрограммы SPLINE и SEVAL. Интерполирование по Эрмиту. Обратная задача интерполирования.	19
8.1 Интерполирование сплайнами.	19
8.2 Подпрограммы SPLINE и SEVAL	20
8.3 Интерполирование по Эрмиту.	21
8.4 Обратная интерполяция.	21

9. Квадратурные формулы левых, правых и средних прямоугольников, трапеций, Симпсона. Малые и составные формулы, их остаточные члены.	22
9.1 Простейшие квадратурные формулы	22
9.2 Погрешность малых квадратурных формул	23
9.3 Составные квадратурные формулы	24
9.4 Погрешности составных квадратурных формул	26
10.Общий подход к построению квадратурных формул. Квадратурные формулы Ньютона-Котеса, Чебышева, Гаусса.	27
10.1 Общий подход к построению квадратурных формул.	27
10.2 Квадратурные формулы Ньютона-Котеса	28
10.3 Квадратурные формулы Чебышева	28
10.4 Квадратурные формулы Гаусса	28
11.Адаптивные квадратурные формулы. Подпрограмма QUANC8.	29
12.Задача численного дифференцирования. Влияние вычислительной погрешности.	31
12.1 Задача численного дифференцирования.	31
12.2 Влияние погрешности задания функции на точность.	33
13.Среднеквадратичная аппроксимация (дискретный случай). Понятие веса.	33
13.1 Дискретный случай. Весовые коэффициенты.	34
14.Среднеквадратичная аппроксимация (непрерывный случай). Понятие ортогональности.	36
15.Ортогонализация по Шмидту. Примеры ортогональных полиномов.	36
16.Обратная матрица, собственные числа и векторы. Задачи на матрицы. Норма матрицы, сходимость матричного степенного ряда, функции от матрицы.	37
17.7 теорем о матричных функциях.	37
18.Решение систем линейных дифференциальных и разностных уравнений с постоянной матрицей.	37
19.Устойчивость решений дифференциальных и разностных уравнений.	37
20.Метод Гаусса и явление плохой обусловленности. LU-разложение матрицы. Подпрограммы DECOMP и SOLVE.	37
21.Метод последовательных приближений для решения линейных систем.	37

22.Методы бисекции, секущих, обратной параболической интерполяции для решения нелинейных уравнений. Подпрограмма ZEROIN.	38
23.Методы последовательных приближений и Ньютона для решения нелинейных уравнений и систем.	38
24.Задача Коши решения обыкновенных дифференциальных уравнений. Явный и неявный методы ломаных Эйлера, метод трапеций.	38
25.Методы Адамса. Локальная и глобальная погрешности, степень метода.	38
26.Методы Рунге-Кутты. Подпрограмма RKF45.	38
27.Глобальная погрешность. Устойчивость метода. Ограничение на шаг. Явление жесткости и методы решения жестких систем.	38
28.Метод Ньютона в неявных алгоритмах решения дифференциальных уравнений.	39
29.Сведение дифференциального уравнения высокого порядка к системе уравнений первого порядка. Метод стрельбы для решения краевых задач.	39

Вопрос 1. Конечные разности и их свойства.

Таблица конечных разностей.

1.1 Конечные разности и их свойства.

Definition 1: Конечная разность

Пусть значения некоторой функции $f(x)$ известны лишь для дискретного множества значений независимой переменной $x \in \{x_0 \dots x_m\}$. Выражение

$$\Delta_h f(x_k) = f(x_k + h) - f(x_k) = f(x_0 + (k+1)h) - f(x_0 + kh) \quad (1)$$

называют *конечной разностью* (*разностным оператором*) первого порядка.

Поскольку величины x_0 и h постоянны для рассматриваемого множества, целесообразно, не умаляя общности, перейти к новой переменной $k = \frac{x_k - x_0}{h}$, которая принимает целые значения $0 \dots m-1$. Тогда функция $f(x)$ становится функцией целочисленной переменной $f(k)$, и можно будет опустить индекс $h = \text{const}$.

$$\Delta f(k) = \Delta f_k = f(k+1) - f(k) = f_{k+1} - f_k$$

Теперь обратимся к некоторым свойствам конечных разностей, отмечая тесную связь между ними и свойствами производных, что является основой большинства конечноразностных выражений.

$$1. \alpha = \text{const} \Rightarrow \Delta \alpha = 0$$

$$\alpha = \text{const} \Rightarrow \forall k \quad f(k+1) - f(k) = \alpha - \alpha = 0$$

$$2. \Delta(\alpha f(k)) = \alpha \Delta f(k)$$

$$3. \Delta(f(k) \pm g(k)) = \Delta f(k) \pm \Delta g(k)$$

$$4. \Delta(f(k) \cdot g(k)) = \Delta f(k) \cdot g(k+1) + f(k) \Delta g(k)$$

$$\begin{aligned} \Delta(f(k) \cdot g(k)) &= f(k+1)g(k+1) - f(k)g(k) = \\ &= f(k+1)g(k+1) - f(k)g(k) + f(k)g(k+1) - f(k)g(k+1) = \\ &= \Delta f(k) \cdot g(k+1) + f(k) \Delta g(k) \end{aligned}$$

Заметим, что аналогичными преобразованиями можно было получить и другой вид, в котором функции идут в другом порядке:

$$\Delta(f(k) \cdot g(k)) = \Delta g(k) \cdot f(k+1) + g(k) \Delta f(k)$$

$$5. \text{ Конечная разность от полинома степени } s \text{ равна полиному степени } s-1.$$

$$\Delta k^s = (k+1)^s - k^s = sk^{s-1} + \frac{s(s-1)}{2}k^{s-2} + \dots$$

6. Конечная разность высокого порядка.

Подобно дифференциалам и производным высокого порядка, соответствующие конечные разности строятся на основе рекуррентных соотношений. Так конечная разность порядка $s + 1$ строится следующим образом:

$$\Delta^{s+1} f_k = \Delta (\Delta^s f_k) = \Delta^s f_{k+1} - \Delta^s f_k$$

По индукции можно доказать следующее утверждение:

Theorem 1: О конечных разностях высокого порядка

$$\Delta^s f_k = \sum_{i=0}^s (-1)^i C_s^i f_{k+s-i} \quad (2)$$

1.2 Таблица конечных разностей.

Аналогично тому, как в непрерывном случае строилась таблица производных, рассмотрим конечные разности для наиболее популярных функций.

$$1. \Delta a^k = a^{k+1} - a^k = a^k(a - 1)$$

Заметим, что число 2 в условиях конечных разностей играет роль, схожую с экспонентой в непрерывном случае: $(e^x)' = e^x$.

$$2. \Delta \sin(k) = \sin(k + 1) - \sin(k) = 2 \sin\left(\frac{1}{2}\right) \cos\left(k + \frac{1}{2}\right)$$

$$3. \Delta \cos(k) = \cos(k + 1) - \cos(k) = -2 \sin\left(\frac{1}{2}\right) \sin\left(k + \frac{1}{2}\right)$$

$$4. \Delta \log(k) = \log(k + 1) - \log(k) = \log\left(1 + \frac{1}{k}\right)$$

Вопрос 2. Суммирование функций. Формула Абеля суммирования по частям.

2.1 Суммирование функций.

Обратимся к уравнению

$$\Delta F(k) = \varphi(k) \quad (3)$$

До сих пор мы занимались прямой задачей: по заданной функции $F(k)$ необходимо определить функцию $\varphi(k)$. Теперь обратимся к обратной задаче: по заданной функции $\varphi(k)$ необходимо восстановить функцию $F(k)$. Ситуация подобно нахождению функции по ее производной в непрерывных терминах. В этом случае появляется возможность ее решения при помощи интеграла

$$\int_a^b h(x) dx = \int_a^b f'(x) dx = f(b) - f(a)$$

Аналогично, решение обратной задачи (3) позволяет, в свою очередь, успешно решать задачу суммирования функции $\varphi(k)$. Запишем уравнение (3) последовательно для $k = m, m + 1, \dots, N - 1$ и результаты просуммируем.

$$\begin{aligned} F(m+1) - F(m) &= \varphi(m) \\ F(m+2) - F(m+1) &= \varphi(m+1) \\ F(m+3) - F(m+2) &= \varphi(m+2) \\ &\dots \\ F(N) - F(N-1) &= \varphi(N-1) \\ F(N) - F(m) &= \sum_{k=m}^{N-1} \varphi(k) \end{aligned}$$

Иными словами

$$\sum_{k=m}^{N-1} \varphi(k) = \sum_{k=m}^{N-1} \Delta F(k) = F(N) - F(m) \quad (4)$$

Выражение (4) является дискретным аналогом формулы Ньютона-Лейбница. В дополнение следует заметить, что она выводилась в предположении, что $N > m$.

Рассмотрим некоторые примеры суммирования функций. Результаты отдаленно напоминают таблицу интегралов для непрерывных функций, а оператор суммы сопоставляется определенному интегралу:

$$\sum_{k=0}^{N-1} \leftrightarrow \int_0^N$$

1. Найти $\sum_{k=0}^{N-1} a^k$

Функция $F(k)$, удовлетворяющая условию $\Delta F(k) = a^k$ легко находится из таблицы конечных разностей: $F(k) = \frac{a^k}{a-1}$. Тогда

$$\sum_{k=0}^{N-1} a^k = \frac{(a^N - 1)}{a - 1} - \frac{a^0 - 1}{a - 1} = \frac{1 - a^N}{1 - a}$$

2. Найти $\sum_{k=0}^{N-1} \cos\left(k + \frac{1}{2}\right)$

Аналогичным образом найдем функцию из таблицы конечных разностей.

$$\sum_{k=0}^{N-1} \cos\left(k + \frac{1}{2}\right) = \sum_{k=0}^{N-1} \frac{\sin(k)}{2 \sin\left(\frac{1}{2}\right)} = \frac{\sin(N)}{2 \sin\left(\frac{1}{2}\right)}$$

3. Найти $\sum_{k=0}^{N-1} k^2$

Воспользуемся свойством №5 конечных разностей: полином степени 2 является конечной разностью полинома степени 3. Рассмотрим

$$\Delta k^3 = (k+1)^3 - k^3 = 3k^2 + 3k + 1$$

$$\Delta k^2 = (k+1)^2 - k^2 = 2k + 1$$

$$\Delta k = (k+1) - k = 1$$

Используя также свойства №1 и 2, получаем

$$\Delta \left(\frac{k^3}{3} - \frac{k^2}{2} + \frac{k}{6} \right) = \frac{1}{3} (3k^2 + 3k + 1) - \frac{1}{2} (2k + 1) + \frac{1}{6} \cdot 1 = k^2 + k + \frac{1}{3} - k - \frac{1}{2} + \frac{1}{6} = k^2$$

Получаем

$$F(k) = \frac{1}{3}k^3 - \frac{1}{2}k^2 + \frac{1}{6}k$$

$$\sum_{k=0}^N k^2 = \frac{N^3}{3} - \frac{N^2}{2} + \frac{N}{6}$$

2.2 Суммирование по частям.

Суммирование по частям вводится как прием, аналогичный интегрированию по частям в непрерывном случае. Для вывода формулы запишем уравнение интегрирования по частям в несколько другом виде. Введем три функции: $u(t), v(t), U(t)$, где

$U(t) = \int_0^t u(\tau) d\tau$. Рассмотрим производную выражения $U(t)v(t)$:

$$\frac{d}{dt} (U(t)v(t)) = \frac{dU(t)}{dt} v(t) + U(t) \frac{dv(t)}{dt} = u(t)v(t) + U(t) \frac{dv(t)}{dt}$$

Перенесем второе слагаемое из правой части равенства в левую и проинтегрируем

$$\int_a^b u(t)v(t) dt = \int_a^b \frac{d}{dt} (U(t)v(t)) dt - \int_a^b U(t) \frac{dv(t)}{dt} dt = U(t)v(t) \Big|_{t=a}^{t=b} - \int_a^b U(t) \frac{dv(t)}{dt} dt$$

Теперь обратимся к суммированию. Аналогично введем три функции: $u(k), v(k), U(k)$,

где $U(k) = \sum_{i=0}^k u(i)$. $\Delta U(k) = U(k+1) - U(k) = u(k+1)$. Воспользуемся ранее полученной формулой конечной разности для произведения:

$$\Delta (U(k)v(k)) = v(k+1)\Delta U(k) + U(k)\Delta v(k) = v(k+1)u(k+1) + U(k)\Delta v(k)$$

Перенесем второе слагаемое из правой части равенства в левую и просуммируем обе его части:

Theorem 2: Формула Абеля суммирования по частям

$$\sum_{k=p}^N u(k+1)v(k+1) = U(k)v(k) \Big|_{k=p}^{k=N+1} - \sum_{k=p}^N U(k)\Delta v(k) \quad (5)$$

Пример: требуется найти $\sum_{k=0}^N ka^k$.

Возьмем функции $u(k) = a^{k-1}$, $v(k) = k - 1$. Тогда

$$U(k) = \sum_{i=0}^k a^{k-1} = \frac{1}{a} \frac{a^{k+1} - 1}{a - 1}$$

$$\Delta v(k) = \Delta k = 1$$

$$\begin{aligned} \sum_{k=0}^N ka^k &= \frac{1}{a} \frac{a^{N+2} - 1}{a - 1} \cdot (((N + 1) - 1) - (0 - 1)) - \sum_{k=0}^N \frac{1}{a} \frac{a^{k+1} - 1}{a - 1} = \\ &= \frac{(N + 1)(a^{N+2} - 1)}{a(a - 1)} - \frac{1}{a(a - 1)} \left(a \sum_{k=0}^N a^k - (N + 1) \right) = \\ &= \frac{(N + 1)(a^{N+2} - 1)}{a(a - 1)} - \frac{1}{a(a - 1)} \left(a \frac{a^{N+1} - 1}{a - 1} - (N + 1) \right) = \\ &= \frac{(a - 1)(N + 1)(a^{N+2} - 1) - a^{N+3} + a^2 + (a - 1)(N + 1)}{a(a - 1)^2} = \\ &= \frac{(a - 1)(N + 1)a^{N+1} - a^{N+2} + a}{(a - 1)^2} = \frac{Na^{N+2} - (N + 1)a^{N+1} + a}{(a - 1)^2} = \\ &= \frac{a(Na^{N+1} - (N + 1)a^N + 1)}{(a - 1)^2} \end{aligned}$$

Вопрос 3. Разностное уравнение, его порядок.

Линейные разностные уравнения первого порядка и порядка выше первого.

3.1 Разностное уравнение, его порядок.

Первоначально обратимся к дифференциальным уравнениям.

Definition 2: Дифференциальное уравнение

Соотношение

$$F(t, z(t), z'(t), \dots, z^{(s-1)}(t)) = 0$$

где t – независимая переменная, функция F задана, функция $z(t)$ – искомая, называется *дифференциальным уравнением порядка s* .

При этом уравнение может быть разрешено относительно старшей производной

$$z^{(s)}(t) = f(t, z(t), z'(t), \dots, z^{(s-1)}(t)) \quad (6)$$

Порядок уравнения s , определяемый номером старшей производной, является важной характеристикой уравнения (6). Так он определяет количество начальных условий, необходимых для однозначного решения. Если дифференциальное уравнение является линейным относительно функции $z(t)$ и ее производных, то величина s задает количество линейно независимых решений и т.д.

Рассмотрим разностный аналог дифференциального уравнения

$$F(k, f(k), \Delta f(k), \Delta^2 f(k), \dots, \Delta^s f(k)) = 0 \quad (7)$$

где k – независимая целочисленная переменная, функция F задана, функция $f(k)$ – искомая. Казалось бы, логично считать порядок этого уравнения равным s , руководствуясь номером старшей конечной разности, как это было с производными в уравнении (6). Рассмотрим, однако, следующий пример:

$$2\Delta^3 f_k + 3\Delta^2 f_k - f_k = 0, \quad f_k \equiv f(k)$$

Выразим все конечные разности через значения функции в различных точках и получим

$$\begin{aligned} 2(f_{k+3} - 3f_{k+2} + 3f_{k+1} - f_k) + 3(f_{k+2} - 2f_{k+1} + f_k) - f_k &= 0, \\ 2f_{k+3} - 3f_{k+2} &= 0 \end{aligned}$$

Задаваясь только одним начальным условием f_0 вместо ожидаемых трех и последовательно полагая значение $k = -2, -1, 0, 1, \dots$ шаг за шагом воспроизводим f_k для любого значения k . В этом и есть различие между дифференциальными и разностными уравнениями. Снижение ожидаемого порядка произошло за счет сокращения слагаемых. По этой причине в общем случае для определения порядка разностного уравнения будем выражать все конечные разности через значения функции. Тогда, после всех упрощений порядок разностного уравнения будет определяться разностью между наибольшим и наименьшим значениями аргумента функции $f(k)$. В дальнейшем будем записывать разностные уравнения в следующем виде.

Definition 3: Разностное уравнение

Уравнение вида

$$\Phi(k, f(k), f(k+1), \dots, f(k+s)) = 0$$

где k – независимая переменная, функция Φ задана, функция $f(k)$ – искомая, называется *разностным уравнением порядка $s = (k+s) - k$* .

или в виде, разрешенном относительно функции с наибольшим значением аргумента

$$f(k+s) = \Phi_1(k, f(k), \dots, f(k+s-1))$$

Для его решения достаточно последовательно полагать $k = 0, 1, 2, \dots$

$$\begin{aligned} f(s) &= \Phi_1(0, f(0), \dots, f(s-1)), \\ f(s+1) &= \Phi_1(1, f(1), \dots, f(s)), \\ f(s+2) &= \Phi_1(2, f(2), \dots, f(s+1)), \\ &\dots \end{aligned}$$

Такое построение решения называют *пошаговым методом решения разностного уравнения*, который всегда дает решение, когда заданы s начальных условий.

3.2 Линейное разностное уравнение первого порядка

Обратимся к уравнению

$$y(k+1) = \alpha \cdot y(k) + \varphi(k), \quad y(0) = y_0 \quad (8)$$

где α – постоянный коэффициент, $\varphi_k = \varphi(k)$ – заданная функция k , $y_k = y(k)$ – искомая функция. Если $\varphi(k) = 0$, уравнение называется однородным, в противном случае – неоднородным. Начнем решать уравнение (8) пошаговым методом.

$$\begin{aligned} y_1 &= \alpha \cdot y_0 + \varphi_0, \\ y_2 &= \alpha \cdot y_1 + \varphi_1 = \alpha(\alpha \cdot y_0 + \varphi_0) + \varphi_1 = \alpha^2 \cdot y_0 + \alpha \cdot \varphi_0 + \varphi_1, \\ y_3 &= \alpha \cdot y_2 + \varphi_2 = \alpha(\alpha^2 \cdot y_0 + \alpha \cdot \varphi_0 + \varphi_1) + \varphi_2 = \alpha^3 \cdot y_0 + \alpha^2 \cdot \varphi_0 + \alpha \cdot \varphi_1 + \varphi_2, \\ &\dots \end{aligned}$$

По индукции можно доказать, что

$$y_n = \alpha^n \cdot y_0 + \sum_{k=0}^{n-1} \alpha^k \cdot \varphi_{n-1-k}$$

Для важного частного случая, когда $\varphi(k)$ – постоянная функция ($\varphi_k = \beta = \text{const}$):

$$y_n = \alpha^n \cdot y_0 + \left(\sum_{k=0}^{n-1} \alpha^k \right) \cdot \beta = \alpha^n \cdot y_0 + \frac{1 - \alpha^n}{1 - \alpha} \cdot \beta$$

3.3 Линейное разностное уравнение порядка выше первого

Перейдем к уравнению порядка s :

$$y(k+s) + \alpha_1 y(k+s-1) + \alpha_2 y(k+s-2) + \dots + \alpha_s y(k) = \varphi(k) \quad (9)$$

где α_i – постоянные коэффициенты, $\varphi(k)$ – заданная функция, $y(k)$ – искомая функция.

Рассмотрим некоторые свойства частных и общих решений систем линейных разностных уравнений.

Theorem 3

Пусть $y_1(k), y_2(k), \dots, y_p(k)$ – частные решения линейного однородного уравнения

$$y(k+s) + \alpha_1 y(k+s-1) + \alpha_2 y(k+s-2) + \dots + \alpha_s y(k) = 0 \quad (10)$$

то любая их линейная комбинация

$$c_1 y_1(k) + c_2 y_2(k) + \dots + c_p y_p(k)$$

где c_i – произвольные постоянные, также будет частным решением этого уравнения.

Theorem 4

Если s частных решений однородного уравнения $y_1(k), y_2(k), \dots, y_s(k)$ – линейно независимы, то

$$y(k) = \sum_{i=1}^s c_i y_i(k) \quad (11)$$

является общим решением однородного уравнения.

Theorem 5

Общее решение линейного неоднородного уравнения (9) представляется в виде суммы частного его решения $y_{\text{частн}}(k)$ и общего решения линейного уравнения (11)

$$y(k) = y_{\text{частн}}(k) + \sum_{i=1}^s c_i y_i(k)$$

Решение неоднородного уравнения (9) начинается с решения однородного уравнения (10). Это решение будем искать в виде $u(k) = C\gamma^k$, где $C = \text{const}$. Здесь уместно вспомнить, что в дифференциальном уравнении порядка s с постоянными коэффициентами частные решения ищутся в форме $z(t) = C \cdot \exp(\lambda_k t)$. Подставим $u(k) = C\gamma^k$ в уравнение (10) и после сокращения получаем уравнение

$$\gamma^s + \alpha_1 \gamma^{s-1} + \dots + \alpha_s = 0 \quad (12)$$

которое получило название *характеристического уравнения*. Оно, с учетом кратности, имеет s корней, каждому из которых соответствует частное решение.

1. Каждому простому вещественному корню γ_r соответствует частное решение $u_r(k) = c_r \gamma_r^k$, являющееся одним из слагаемых в общем решении.
2. Каждой простой паре комплексно-сопряженных корней $\gamma_{r,r+1} = (\alpha_r \pm i\beta_r)$ соответствуют комплексные частные решения, являющиеся линейно независимыми

$$u_r(k) = (\alpha_r + i\beta_r)^k, \quad u_{r+1}(k) = (\alpha_r - i\beta_r)^k$$

или вещественные частные решения

$$u_r(k) = \rho_r^k \cos(k\varphi_r), \quad u_{r+1}(k) = \rho_r^k \sin(k\varphi_r), \quad \rho_r = \sqrt{\alpha_r^2 + \beta_r^2}, \quad \text{tg}(\varphi_r) = \frac{\beta_r}{\alpha_r}$$

В общем решении им сопоставляются два слагаемых (вещественный вариант)

$$c_r \rho_r^k \cos(k\varphi_r) + c_{r+1} \rho_r^k \sin(k\varphi_r)$$

3. Если среди корней встречаются кратные, то корню γ_r кратности p соответствуют частные решения

$$u_r(k) = \gamma_r^k, \quad u_{r+1}(k) = k\gamma_r^k, \dots, u_{r+p-1}(k) = k^{p-1}\gamma_r^k$$

Решения эти линейно зависимы, и в общем решении им сопоставляются слагаемые

$$c_r \gamma_r^k + c_{r+1} k \gamma_r^k + \dots + c_{r+p-1} k^{p-1} \gamma_r^k = Q_{p-1}(k) \gamma_r^k$$

где $Q_{p-1}(k)$ – полином от k степени $p-1$.

Вопрос 4. Разделенные разности и их связь с конечными разностями.

Для равноотстоящих узлов таблицы конечные разности являются хорошей характеристикой изменения функции, аналогичной производной для непрерывного случая. При произвольном расположении узлов таблицы целесообразно ввести понятие *разделенной разности*.

Definition 4: Разделенная разность

Разделенные разности нулевого порядка совпадают со значениями функции, а разности первого порядка определяются равенством

$$f(x_{n-1}; x_n) = \frac{f(x_n) - f(x_{n-1})}{x_n - x_{n-1}} \quad (13)$$

Аналогично строятся разделенные разности высших порядков. При этом разности k -го порядка определяются через разности $(k-1)$ -го порядка по формуле

$$f(x_0; x_1; \dots; x_k) = \frac{f(x_1; x_2; \dots; x_k) - f(x_0; x_1; \dots; x_{k-1})}{x_k - x_0} \quad (14)$$

Подобно конечным разностям, разделенные тоже можно выразить через значения функции в различных точках. По индукции можно доказать следующее равенство:

$$f(x_i; x_{i+1}; \dots; x_{i+k}) = \sum_{j=i}^{i+k} \frac{f(x_j)}{\prod_{j \neq i} (x_j - x_i)} \quad (15)$$

Отсюда следует важное свойство разделенных разностей: они являются симметричными функциями своих аргументов.

$$f(x_n; x_{n-1}) = f(x_{n-1}; x_n)$$

Если в исходной таблице узлы равноотстоящие, то для описания поведения функции можно использовать как конечные разности, так и разделенные. Установим связь между ними. Обобщенную формулу можно доказать по индукции.

$$f(x_i; x_{i+1}) = \frac{f(x_{i+1}) - f(x_i)}{x_{i+1} - x_i} = \frac{\Delta f_i}{h} \quad \dots$$

Theorem 6: Связь конечных и разделенных разностей

$$f(x_i; x_{i+1}; \dots; x_{i+k-1}; x_{i+k}) = \frac{\Delta^k f_i}{k! h^k} \quad (16)$$

Вопрос 5. Аппроксимация функций. Задача интерполирования.

5.1 Аппроксимация функций.

В данной теме аппроксимация означает *замену одной функциональной зависимости другой*. Поскольку на практике часто возникает потребность дифференцировать, интегрировать или использовать эту функцию в различных расчетах, целесообразно выбирать аппроксимирующую функцию, исходя из простоты ее вида. Возможность выбора обосновывается следующей теоремой.

Theorem 7: Теорема Вейерштрасса

Пусть $f(x)$ непрерывна на $[a, b]$. Тогда

$$\forall \varepsilon > 0 \quad \exists P_n(x) \quad n = n(\varepsilon) : \quad \max_{x \in [a, b]} |f(x) - P_n(x)| < \varepsilon$$

Однако, эта теорема лишь гарантирует существование, но не дает гарантии, что такой полином можно построить при помощи практического алгоритма.

Для того, чтобы можно было сравнивать различные варианты аппроксимации, следует ввести критерий близости. Например, максимум модуля отклонения исходной функции $f(x)$ от аппроксимирующей $g(x)$ на заданном промежутке:

$$\delta = \max_{x \in [a, b]} |f(x) - g(x)| \quad (17)$$

или так называемый «среднеквадратичный критерий»

$$\rho^2 = \int_a^b (f(x) - g(x))^2 dx \quad (18)$$

В случае если $f(x)$ определена таблично заданным набором точек, может быть использован аналог критерия (18)

$$\rho^2 = \sum_{k=1}^m (f(x_k) - g(x_k))^2 \quad (19)$$

Лучшей оказывается аппроксимирующая функция, обладающая наименьшей величиной δ или ρ^2 . Заметим, что только решаемая задача диктует выбираемый критерий близости, который, в свою очередь, позволяет выбрать лучшую аппроксимацию.

5.2 Постановка задачи интерполирования

Будем приближать исходную функцию, заданную таблично $X = \{x_0, x_1, \dots, x_m\}$, $F = \{f(x) \mid x \in X\}$ обобщенным многочленом

$$Q_m(x) = a_0\varphi_0(x) + a_1\varphi_1(x) + \dots + a_m\varphi_m(x) = \sum_{k=0}^m a_k\varphi_k(x) \quad (20)$$

где $\{\varphi_k\}$ – заданный набор линейно независимых функций, а коэффициенты a_i подлежат определению. В качестве критерия близости выбирается совпадение значений $f(x)$ и $Q_m(x)$ в узлах таблицы

$$Q_m(x_i) = f(x_i), \quad i = 0, 1, \dots, m \quad (21)$$

Definition 5: Интерполяционный многочлен

Полином (20) называется *интерполяционным многочленом*, а x_k – *узлами интерполирования*.

Равенства (21) представляют собой СЛАУ относительно искомых коэффициентов обобщенного многочлена a_0, a_1, \dots, a_m . Эта система имеет единственное решение, если ее определитель отличен от нуля:

$$\det \begin{pmatrix} \varphi_0(x_0) & \varphi_1(x_0) & \dots & \varphi_m(x_0) \\ \varphi_0(x_1) & \varphi_1(x_1) & \dots & \varphi_m(x_1) \\ \dots & \dots & \dots & \dots \\ \varphi_0(x_m) & \varphi_1(x_m) & \dots & \varphi_m(x_m) \end{pmatrix} \neq 0$$

Наиболее популярной является полиномиальная аппроксимация:

$$\varphi_k(x) = x^k \quad Q_m(x) = a_0 + a_1x + a_2x^2 + \dots + a_mx^m$$

Definition 6: Определитель Вандермонда

Определитель СЛАУ для случая полиномиальной аппроксимации называется *определителем Вандермонда* и имеет следующий вид:

$$\begin{vmatrix} 1 & x_0 & \dots & x_0^n \\ 1 & x_1 & \dots & x_1^n \\ \dots & \dots & \dots & \dots \\ 1 & x_n & \dots & x_n^n \end{vmatrix}$$

Определитель Вандермонда отличен от нуля, и задача имеет единственное решение, если узлы интерполирования x_0, x_1, \dots, x_m различны.

Вопрос 6. Интерполяционный полином Лагранжа.

Остаточный член полинома Лагранжа.

Непосредственное численное решение представляет значительные трудности. С одной стороны, это связано с заметным объемом вычислений для нахождения a_k . С другой стороны, малое изменение данных таблицы $(x_k, f(x_k))$ часто приводит к сильному изменению решения (особенно для близко расположенных узлов интерполирования). В связи с этим, попробуем построить полином, не прибегая к решению системы. С этой целью введем следующие функции:

$$\omega(x) = (x - x_0)(x - x_1) \dots (x - x_m)$$

$$\omega_k(x) = \frac{\omega(x)}{(x - x_k)} = (x - x_0)(x - x_1) \dots (x - x_{k-1})(x - x_{k+1}) \dots (x - x_m)$$

В этих обозначениях запишем следующий полином

Definition 7: Интерполяционный полином Лагранжа

$$Q_m(x) = \sum_{k=0}^m \frac{\omega_k(x)}{\omega_k(x_k)} f(x_k) \quad (22)$$

По построению это многочлен степени m . Определим его значения в узлах интерполирования x_i . Так как для $x = x_i$ полином $\omega_k(x)$ равен нулю, если только $i \neq k$, то для $Q_m(x_i)$ получаем

$$Q_m(x_i) = \sum_{k=0}^m \frac{\omega_k(x_i)}{\omega_k(x_k)} f(x_k) = \frac{\omega_k(x_i)}{\omega_k(x_i)} f(x_i) = f(x_i), \quad i = 0, 1, \dots, m$$

То есть в узлах интерполяции значения полинома совпадают со значениями функции. Теперь обратимся к погрешности интерполяционного полинома. Исходная функция $f(x)$ может быть представлена в виде

$$f(x) = Q_m(x) + R_m(x)$$

где $Q_m(x)$ – интерполяционный полином, а $R_m(x)$ носит название *остаточного члена интерполяционного полинома*.

Theorem 8: Об остаточном члене полинома Лагранжа

Пусть $f(x)$ на промежутке $[a, b]$ имеет непрерывные производные вплоть до $m+1$ порядка, то остаточный член $R_m(x)$ можно представить в виде:

$$R_m(x) = f(x) - Q_m(x) = \frac{f^{(m+1)}(\eta)}{(m+1)!} \omega(x), \quad \eta \in [a, b] \quad (23)$$

При этом $\omega(x)$ определяется как и прежде.

Доказательство. Рассмотрим вспомогательную функцию

$$\varphi(z) = f(z) - Q_m(z) - K\omega(z) \quad (24)$$

где K – некоторая постоянная. Пусть x_k – узлы интерполирования, а x – точка, в которой оценивается погрешность ($x \neq x_k$). Легко заметить, что функция $\varphi(z)$ равна нулю во всех узлах интерполирования. Выберем константу K так, чтобы $\varphi(x) = 0$

$$K = \frac{f(x) - Q_m(x)}{\omega(x)} = \frac{R_m(x)}{\omega(x)}$$

Таким образом, $\varphi(z)$ имеет по меньшей мере $m + 2$ нуля (все узлы интерполирования и точка x). Тогда по теореме Ролля первая производная $\varphi(z)$ имеет по меньшей мере $m + 1$ нуль, вторая производная – не менее m нулей, а $(m + 1)$ -я производная $\varphi^{(m+1)}(z)$ имеет по меньшей мере один нуль. Обозначим такую точку за η . Тогда, последовательно дифференцируя (24), получаем

$$\varphi^{(m+1)}(\eta) = f^{(m+1)}(\eta) - 0 - K(m + 1)! = 0$$

Подставляя в это равенство выражение для K , получаем формулу для $R_m(x)$, совпадающую с ожидаемой.

Эта теорема позволяет сделать очевидный, но важный вывод. Пусть $f(x)$ – это полином степени m . Тогда $f^{(m+1)}(\eta) = 0$. Следовательно, полином степени m *однозначно* воспроизводится интерполяционным полиномом по $m + 1$ точке. Ясно также, что остаточный член во всех узлах интерполирования равен нулю.

В заключение стоит отметить, что, хотя о расположении точки η ничего не известно, очевидна зависимость величины η как от узлов интерполирования, так и от точки x , где оценивается погрешность, т.е. $\eta = \eta(x)$.

Остаточный член позволяет оценивать отклонение $L_m(x)$ от $f(x)$ для дифференцируемых функций тогда, когда удастся оценить $f^{(m+1)}(x)$. Полагая $M_{m+1} = \max |f^{(m+1)}(x)|$, получим $R_m(x) \leq \frac{M_{m+1}}{(m + 1)!} |\omega(x)|$.

Вопрос 7. Выбор узлов интерполирования.

Интерполяционный полином Ньютона для равно и неравноотстоящих узлов.

7.1 Выбор узлов интерполирования

Для уменьшения погрешности интерполирования обратимся к теореме об остаточном члене полинома Лагранжа при заданной степени полинома m . Поскольку величиной $f^{(m+1)}(\eta)$ трудно управлять, и возможна лишь оценка пределов ее изменения, задача уменьшения погрешности сводится к управлению величиной $|\omega(x)|$ за счет выбора узлов интерполирования. Рассмотрим два типичных на практике случая.

Случай 1. Задана степень полинома m и имеется таблица достаточно большой длины. Точка x^* , в которой вычисляется значение полинома, заранее известна. Требуется выбрать $m + 1$ узел так, чтобы величина $|\omega(x^*)|$ была бы минимальна.

Результат очевиден. Нужно выбирать узлы интерполирования из таблицы, *ближайшие* к x^* . Использование любого другого узла вместо ближайшего неизбежно увеличивает значение

$$|\omega(x^*)| = |(x^* - x_0)(x^* - x_1) \dots (x^* - x_m)|$$

Случай 2. Заданы степень полинома m и промежуток интерполирования $[a, b]$. Точка x^* , в которой вычисляется значение полинома, заранее не известна. Требуется выбрать узлы интерполирования так, чтобы в самом неблагоприятном случае расположения x^* погрешность была бы минимальна (т.н. *минимаксный критерий*)

$$\max_{[a,b]} |\omega(x^*)| \rightarrow \min$$

Интуитивно напрашивающееся предложение о равномерном задании узлов на промежутке оказывается ошибочным. Значения $|\omega(x)|$ в узлах интерполирования равны нулю, график напоминает «колокольчики», максимум которых достигается между узлами интерполирования. При выборе равноотстоящих узлов погрешность для x^* , близких к центру промежутка интерполирования оказывается небольшой, однако ближе к концам она сильно возрастает. Узлы интерполирования нужно симметрично сместить ближе к концам промежутка. Тогда высота центрального «колокольчика» увеличится, в то время как высота крайних уменьшится. Оптимальный выбор узлов интерполирования отвечает нулям так называемых ортогональных полиномов Чебышева, когда все «колокольчики» будут одинаковыми по высоте.

7.2 Интерполяционный полином Ньютона для равно и неравноотстоящих узлов.

Оценка погрешности на основе формулы

$$R_m(x) = \frac{f^{(m+1)}(\eta)}{(m+1)!} \omega(x)$$

выполняется крайне редко из-за известных трудностей, связанных с оценкой производной $f^{(m+1)}(\eta)$ особенно для таблично заданной функции. Поэтому на практике о величине погрешности принято судить, сравнивая в заданной точке x^* значения интерполяционных полиномов соседних степеней $Q_m(x^*)$ и $Q_{m+1}(x^*)$. При недостаточной точности последовательно повышают степень полинома. Но для такой процедуры использование полинома Лагранжа оказывается неэффективным. При переходе к полиному следующей степени всю работу приходится выполнять заново. Целесообразно записать полином в таком виде, чтобы расчеты сводились к появлению лишь еще одного слагаемого в дополнение к ранее вычисленному $Q_m(x)$. С этой целью запишем первую разделенную разность

$$f(x; x_0) = \frac{f(x) - f(x_0)}{x - x_0}$$

и выразим из нее $f(x)$

$$f(x) = f(x_0) + (x - x_0)f(x; x_0) \quad (25)$$

Заметим, что первое слагаемое в первой части это интерполяционный полином нулевой степени, а второе слагаемое – погрешность полинома. Теперь запишем вторую разделенную разность

$$f(x; x_0; x_1) = \frac{f(x; x_0) - f(x_0; x_1)}{x - x_1}$$

выразим из нее первую разность через вторую и поставим в формулу (25)

$$f(x) = f(x_0) + (x - x_0)f(x_1; x_0) + (x - x_0)(x - x_1)f(x; x_0; x_1) \quad (26)$$

Продолжая этот процесс, получаем

$$\begin{aligned} f(x) &= f(x_0) + (x - x_0)f(x_1; x_0) + (x - x_0)(x - x_1)f(x_0; x_1; x_2) + \dots + \\ &\quad + (x - x_0)(x - x_1) \dots (x - x_{m-1})f(x_0; x_1; x_2; \dots x_m) + \\ &\quad + (x - x_0)(x - x_1) \dots (x - x_m)f(x; x_0; x_1; x_2; \dots x_m) = \\ &= Q_m(x) + \omega(x)f(x; x_0; x_1; x_2; \dots x_m) \end{aligned}$$

Сумма первых k слагаемых порождает интерполяционный полином степени $k - 1$, а последнее слагаемое является погрешностью интерполяционного полинома степени m . При этом структура полинома такова, что полином степени m получается как полином степени $m - 1$ с добавлением еще одного слагаемого.

Definition 8: Интерполяционный полином Ньютона

Это интерполяционный полином в форме

$$Q_m(x) = Q_{m-1}(x) + (x - x_0)(x - x_1) \dots (x - x_{m-1})f(x_0; x_1; x_2; \dots x_m) \quad (27)$$

На практике вычисление разделенных разностей производится в рамках следующей таблицы, где появление новой разделенной разности более высокого порядка связано с построением еще одной диагонали

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

Пусть узлы таблицы задания функции являются равноотстоящими. В этом случае разделенные разности можно заменить на конечные, тем самым избежав деления на разность значений аргумента. Используя

$$\frac{x - x_k}{h} = \frac{x - x_0 - kh}{h} = \frac{x - x_0}{h} - k = t - k$$

и формулу связи разделенных и конечных разностей

$$f(x_i; x_{i+1}; \dots; x_{i+k-1}; x_{i+k}) = \frac{\Delta^k f_i}{k! h^k}$$

получим версию полинома Ньютона для равноотстоящих узлов

$$Q_m(x_0 + ht) = f(x_0) + \frac{t}{1!} \Delta f(x_0) + \frac{t(t-1)}{2!} \Delta^2 f(x_0) + \dots + \frac{t(t-1) \dots (t-m+1)}{m!} \Delta^m f(x_0) \quad (28)$$

Новая независимая переменная t принимает в узлах таблицы целые значения, а вычисление конечных разностей реализуется подобно разделенным разностям по следующей таблице

x_0	$f(x_0)$	Δf_0	$\Delta^2 f_0$	$\Delta^3 f_0$	$\Delta^4 f_0$
x_1	$f(x_1)$	Δf_1	$\Delta^2 f_1$	$\Delta^3 f_1$	
x_2	$f(x_2)$	Δf_2	$\Delta^2 f_2$		
x_3	$f(x_3)$	Δf_3			
x_4	$f(x_4)$				

Вопрос 8. Сплайн-интерполяция. Подпрограммы SPLINE и SEVAL. Интерполирование по Эрмиту. Обратная задача интерполирования.

8.1 Интерполирование сплайнами.

На практике интерполяционные полиномы высоких степеней строят крайне редко. Это связано с тем, что их коэффициенты крайне чувствительны к погрешностям исходных данных. Сравнительно малое изменение узлов интерполирования x_k или значений функции $f(x_k)$ приводит к сильному изменению вида самого полинома. Одним из возможных решений является разбиение большой исходной таблицы на участки, для каждого из которых строится интерполяционный полином относительно невысокой степени. Однако, в основном требуется, чтобы аппроксимирующая функция была гладкой, а функция, составленная из различных полиномов, в узлах сопряжения не имеет производной. Выходом из положения является использование сплайн-интерполяции. Вообще, сплайн – это некий инструмент, используемый при построении чертежей. Дадим математической модели более формальное определение.

Обратимся к таблично заданной функции: $X = \{x_1, \dots, x_N\}$, $F = \{f(x) | x \in X\}$. Число узлов равно N , а их нумерация начинается с единицы. На каждом промежутке $[x_k, x_{k+1}]$ будем строить интерполяционный полином третьей степени

$$S_k(x_{k+1}) = a_k + b_k(x - x_k) + c_k(x - x_k)^2 + d_k(x - x_k)^3 \quad (29)$$

Количество полиномов, как и промежутков, равно $N - 1$, и каждый полином имеет 4 параметра. Таким образом, всего в наличии $4N - 4$ параметра. Потребуем, чтобы во всех внутренних точках были равны значения соседних полиномов, их первых и

вторых производных.

$$\begin{cases} S_k(x_{k+1}) = S_{k+1}(x_{k+1}) \\ S'_k(x_{k+1}) = S'_{k+1}(x_{k+1}) \\ S''_k(x_{k+1}) = S''_{k+1}(x_{k+1}) \end{cases} \quad k = 1, \dots, N-2$$

То есть выполнялось суммарно $3(N-2) = 3N-6$ уравнений. Еще N уравнений отражают требования интерполирования

$$S_k(x_k) = f_k; \quad k = 1, \dots, N-1; \quad S_{N-1}(x_N) = f_N$$

Общее число задаваемых уравнений достигает $4N-6$. При наличии $4N-4$ параметров появляется возможность выполнить еще два условия. Их задание необязательно – все требования интерполяции и сопряжения соседних полиномов уже выполнены, но это целесообразно сделать для однозначного решения задачи. Различные кубические сплайны отличаются друг от друга этим заданием этих двух требований, которые, как правило, записываются для двух крайних точек x_1 и x_N . К этим двум дополнительным условиям целесообразно выдвинуть следующие два требования. С одной стороны, их лучше задавать так, чтобы полная система уравнений решалась по возможности более просто. С другой стороны, они должны максимально соответствовать характеру поведения функции в начале и в конце промежутка интерполирования. Рассмотрим на примерах.

Пример 1. $S''_1(x_1) = 0$; $S''_{N-1}(x_N) = 0$. Этот сплайн получил название *естественного кубического сплайна*. Такие условия и название оправдываются только при использовании в механике. В общем случае равенство нулю второй производной на краях промежутка не является обязательным свойством экспериментальных данных, отражаемых таблицей. *Пример 2.* По первым четырем точкам таблицы строится интерполяционный полином третьей степени $Q_3(x)$, и его третья производная приравнивается третьей производной $S_1(x)$. Аналогично, по последним четырем точкам строится интерполяционный полином $\tilde{Q}_3(x)$, и его третья производная приравнивается третьей производной последнего полинома $S_{N-1}(x)$.

$$Q_3'''(x_1) = S_1'''(x_1); \quad \tilde{Q}_3'''(x_N) = S_{N-1}'''(x_N)$$

Такие условия не только отвечают характеру поведения функции в начале и в конце промежутка интерполирования, но и достаточно просты (третья производная от полинома третьей степени постоянна). Именно они и учитываются в рассматриваемых программах SPLINE и SEVAL.

8.2 Подпрограммы SPLINE и SEVAL.

Первая из них – SPLINE(N, X, F, B, C, D), оформленная как процедура, решает систему уравнений относительно b_k, c_k, d_k .

N – число точек;

X, F – векторы, элементами которых являются x_k и f_k ;

B, C, D – векторы с коэффициентами b_k, c_k, d_k полиномов (29) – результаты работы SPLINE.

Вторая программа $\text{SEVAL}(N, U, X, F, B, C, D)$, оформленная как функция, использует результаты работы SPLINE и вычисляет значение сплайна в заданной точке U .

8.3 Интерполирование по Эрмиту.

До этого интерполяция происходила по значениям функции. Если в таблице помимо значений функции присутствуют ее производные и от интерполяционного полинома требуется совпадение с данными этой таблицы, то такая задача называется *интерполирование по Эрмиту*. Рассмотрим пример.

Для следующих входных данных требуется построить интерполяционный полином, удовлетворяющий всем условиям таблицы.

x	x_0	x_1	x_2
$f(x)$	$f(x_0)$	$f(x_1)$	$f(x_2)$
$f'(x)$	$f'(x_0)$	$f'(x_1)$	—
$f''(x)$	—	$f''(x_1)$	—

Выпишем таблицу в виде системы уравнений:

$$\begin{cases} H(x_k) = f(x_k), & k = 0, 1, 2 \\ H'(x_k) = f'(x_k), & k = 0, 1 \\ H''(x_k) = f''(x_k), & k = 1 \end{cases} \quad (30)$$

Система (30) содержит 6 уравнений. Для ее однозначного решения полином $H(x)$ должен иметь 6 коэффициентов, т.е. быть полиномом пятой степени. Общее правило очевидно: степень интерполяционного полинома Эрмита на единицу меньше общего числа условий таблицы. Вспоминая случай с полиномом Лагранжа, для построения которого решение системы оказалось необязательным, возникает вопрос – нельзя ли и полином Эрмита воспроизвести сразу в готовом виде? Ответ оказывается положительным, однако общая формула довольно громоздка. Форма записи будет проще, если исходная система симметрична, т.е. число и вид условий во всех узлах одинаковые.

Полином Эрмита второй степени

$$H_2(x) = f(x_0) + \frac{x - x_0}{1!} f'(x_0) + \frac{(x - x_0)^2}{2!} f''(x_0)$$

с одной стороны является частичной суммой ряда Тейлора, а с другой удовлетворяет условиям

$$H_2(x_0) = f(x_0); \quad H_2'(x_0) = f'(x_0); \quad H_2''(x_0) = f''(x_0)$$

что позволяет назвать его еще и интерполяционным полиномом Эрмита с одним узлом интерполирования.

8.4 Обратная интерполяция.

До этого была рассмотрена так называемая *прямая* задача интерполирования, в рамках которой по заданному значению x^* требовалось оценить значение функции $f(x^*)$.

В обратной же задаче для такой же таблицы требуется восстановить такое значение аргумента, при котором функция принимает заданное значение. На практике чаще всего используется один из следующих способов.

Способ 1. Меняются местами строки таблицы, в качестве узлов интерполирования выступают значения функции, по которым строится интерполяционный полином для обратной функции. Подставляя в него данное f^* находим искомым x^* . Такой подход возможен, если обратная функция на заданном участке интерполирования существует, то есть исходная функция строго монотонна, что бывает совсем не всегда.

Способ 2. По исходной таблице строится обычный интерполяционный полином $Q_m(x)$ с узлами x_k , а затем решается уравнение $Q_m(x) = f^*$. Для полиномов до 4 степени ответ может быть получен даже аналитически, а в других случаях это уравнение решается численно. В случае немонотонной функции, краевом для предыдущего способа, в этот раз будет найдено несколько корней, из которых необходимо будет выбрать отвечающий поставленной задаче.

Вопрос 9. Квадратурные формулы левых, правых и средних прямоугольников, трапеций, Симпсона. Малые и составные формулы, их остаточные члены.

9.1 Простейшие квадратурные формулы

Definition 9: Квадратурные формулы

Это формулы для вычисления значения определенного интеграла. Их получение – одно из многочисленных возможных приложений интерполяционных полиномов.

Пусть требуется вычислить некий интеграл

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

точное значение которого определить весьма сложно или невозможно. Тогда исходная функция может быть аппроксимирована интерполяционным полиномом $f(x) = Q_m(x) + R_m(x)$ и интеграл от интерполяционного полинома порождает некоторую квадратурную формулу

$$I = \int_a^b f(x)dx \approx \int_a^b Q_m(x)dx \quad (31)$$

а интеграл от остаточного члена определяет ее погрешность

$$\varepsilon = \int_a^b R_m(x)dx \quad (32)$$

Будем последовательно подставлять в (31) полиномы различных степеней, начиная с нулевой ($Q_0(x) = f(x_0)$) $I \approx (b-a)f(x_0)$. Наиболее популярными являются следующие три варианта выбора узла x_0 :

1. Формула левых прямоугольников

$$x_0 = a, \quad I \approx (b-a)f(a) \quad (33)$$

2. Формула правых прямоугольников

$$x_0 = b, \quad I \approx (b-a)f(b) \quad (34)$$

3. Формула средних прямоугольников

$$x_0 = \frac{a+b}{2}, \quad I \approx (b-a)f\left(\frac{a+b}{2}\right) \quad (35)$$

Причины таких названий легко понять из геометрических иллюстраций, откуда видно, что площадь под заданной функцией аппроксимируется площадью соответствующего прямоугольника.

Интегрирование полинома первой степени с узлами интерполирования $x_0 = a$ и $x_1 = b$

$$Q_1(x) = \frac{x-b}{a-b}f(a) + \frac{x-a}{b-a}f(b)$$

порождает квадратурную *формулу трапеций*

$$I \approx \frac{b-a}{2} (f(a) + f(b)) \quad (36)$$

а интегрирование полинома второй степени с узлами $x_0 = a, x_1 = \frac{a+b}{2}, x_2 = b$ приводит к квадратурной *формуле Симпсона*

$$I \approx \frac{b-a}{6} \left(f(a) + 4f\left(\frac{a+b}{2}\right) + f(b) \right) \quad (37)$$

9.2 Погрешность малых квадратурных формул

Оценку погрешности будем выполнять на основе двух теорем о средних:

Theorem 9

1. Пусть $f(x)$ и $g(x)$ непрерывны на $[a, b]$, а $g(x)$ еще и знакопостоянна. Тогда найдется такая точка c , что

$$\int_a^b g(x)f(x)dx = f(c) \int_a^b g(x)dx$$

2. Пусть $f(x)$ непрерывна на $[a, b]$, и заданы N точек $x_k \in [a, b]$. Найдется

точка $\eta \in [a, b]$ такая, что

$$\frac{1}{N} \sum_{k=1}^N f(x_k) = f(\eta)$$

Остаточный член интерполяционного полинома нулевой степени имеет вид

$$R_0(x) = \frac{x - x_0}{1!} f'(\eta)$$

Полезно заметить, что точка η зависит от x , т.е. $\eta = \eta(x)$, как это уже отмечалось при выводе остаточного члена интерполяционного полинома.

Последовательно подставляя значения x_0 , вычислим интеграл (32):

$$\varepsilon_{лев. пр.} = \int_a^b (x - a) f'(\eta) dx = f'(\eta^*) \int_a^b (x - a) dx = \frac{(b - a)^2}{2} f'(\eta^*) \quad (38)$$

$$\varepsilon_{прав. пр.} = \int_a^b (x - b) f'(\eta) dx = f'(\eta^*) \int_a^b (x - b) dx = -\frac{(b - a)^2}{2} f'(\eta^*) \quad (39)$$

Для формулы средних прямоугольников все немного не так, ведь $\left(x - \frac{a+b}{2}\right)$ меняет знак. Для оценки погрешности в этом случае воспользуемся разложением $f(x)$ в ряд в точке $\frac{a+b}{2} = t$

$$f(x) = f(t) + \frac{x - t}{1!} f'(t) + \frac{(x - t)^2}{2!} f''(t)$$

интегрируя его на $[a, b]$. Интеграл от первого слагаемого дает формулу средних прямоугольников, от второго равен нулю, а третий определяет погрешность

$$\varepsilon_{ср. пр.} = \frac{1}{2} \int_a^b \left(x - \frac{a+b}{2}\right)^2 f''(\eta) dx = \frac{f''(\eta^*)}{2} \int_a^b \left(x - \frac{a+b}{2}\right)^2 dx = \frac{(b-a)^3}{24} f''(\eta^*) \quad (40)$$

Для оценки погрешности формулы трапеций проинтегрируем остаточный член полинома первой степени

$$\varepsilon_{trap} = \int_a^b \frac{(x-a)(x-b)}{2!} f''(\eta) dx = f''(\eta^*) \int_a^b \frac{(x-a)(x-b)}{2} dx = -\frac{(b-a)^3}{12} f''(\eta^*) \quad (41)$$

Для интеграла от остаточного члена полинома второй степени условия теоремы о среднем не выполняются, и погрешность формулы Симпсона определяется по другому. По этим же трем узлам строится полинома Эрмита уже третьей степени с двумя условиями в центральной точке, погрешность которого и интегрируется.

$$\varepsilon_{Симпс} = -\frac{(b-a)^5}{2880} f^{(4)}(\eta) \quad (42)$$

9.3 Составные квадратурные формулы

Definition 10

В большинстве случаев подынтегральная функция не описывается удовлетворительно полиномами первой и второй степени. Поэтому, для достижения необходимой точности исходный промежуток разбивается на такие малые промежутки, где указанная аппроксимация удачна, на каждом из этих промежутков применяется выбранная квадратурная формула, а результаты складываются. Такие формулы получили название *составных* к.ф.

Разобьем исходный промежуток $[a, b]$ на N равных промежутков $[x_k, x_{k+1}]$

$$h = \frac{b-a}{N}, \quad x_k = x_0 + kh, \quad x_0 = a, \quad x_N = b$$

На каждом промежутке применим квадратурную формулу и сложим:

1. Левые прямоугольники

$$I_k = \int_{x_k}^{x_{k+1}} f(x) dx \approx (x_{k+1} - x_k) f(x_k) = \frac{b-a}{N} f(x_k)$$

$$I_{лев. пр.} = \sum_{k=0}^{N-1} I_k \approx \frac{b-a}{N} \sum_{k=0}^{N-1} f(x_k)$$

2. Правые прямоугольники

$$I_k = \int_{x_k}^{x_{k+1}} f(x) dx \approx (x_{k+1} - x_k) f(x_{k+1}) = \frac{b-a}{N} f(x_{k+1})$$

$$I_{прав. пр.} = \sum_{k=0}^{N-1} I_k \approx \frac{b-a}{N} \sum_{k=0}^{N-1} f(x_{k+1}) = \frac{b-a}{N} \sum_{k=1}^N f(x_k)$$

3. Средние прямоугольники

$$I_k = \int_{x_k}^{x_{k+1}} f(x) dx \approx (x_{k+1} - x_k) f(x_k + h/2) = \frac{b-a}{N} f(x_k + h/2)$$

$$I_{сред. пр.} = \sum_{k=0}^{N-1} I_k \approx \frac{b-a}{N} \sum_{k=0}^{N-1} f(x_k + h/2) = \frac{b-a}{N} \sum_{k=0}^{N-1} f\left(x_k + \frac{b-a}{2N}\right)$$

4. Трапеции

$$I_k = \int_{x_k}^{x_{k+1}} f(x) dx \approx \frac{x_{k+1} - x_k}{2} (f(x_{k+1}) + f(x_k)) = \frac{b-a}{2N} (f(x_{k+1}) + f(x_k))$$

$$I_{трап.} = \sum_{k=0}^{N-1} I_k \approx \frac{b-a}{2N} \sum_{k=0}^{N-1} (f(x_{k+1}) + f(x_k)) = \frac{b-a}{2N} \left(f(a) + 2 \sum_{k=1}^{N-1} f(x_k) + f(b) \right)$$

Для получения составной формулы Симпсона будем выбирать N всегда четным, а исходный промежуток разобьем на $N/2$ равных промежутков $[x_k, x_{k+2}]$ длиной $2(b-a)/N$. На каждом из них интеграл равен

$$I_k = \int_{x_k}^{x_{k+2}} f(x) dx \approx \frac{x_{k+2} - x_k}{6} (f(x_{k+2}) + 4f(x_{k+1}) + f(x_k)) =$$

$$= \frac{b-a}{3N} (f(x_{k+2}) + 4f(x_{k+1}) + f(x_k))$$

Суммируя по всем промежуткам, получаем *составную формулу Симпсона*

$$I_{\text{Симпс}} = \frac{b-a}{3N} [f(a) + 4(f_1 + f_3 + \dots + f_{N-1}) + 2(f_2 + f_4 + \dots + f_{N-2}) + f(b)] \quad (43)$$

9.4 Погрешности составных квадратурных формул

Для оценки погрешностей составных формул вычисляем погрешность каждого участка и складываем.

1. Левые прямоугольники

$$\begin{aligned} \varepsilon_k &= \frac{(x_{k+1} - x_k)^2}{2} f'(\eta_k) = \frac{(b-a)^2}{2N^2} f'(\eta_k) \\ \varepsilon_{\text{лев. пр.}} &= \sum_{k=1}^N \varepsilon_k = \frac{(b-a)^2}{2N} \left[\frac{1}{N} \sum_{k=1}^N f'(\eta_k) \right] = \frac{(b-a)^3}{24N^2} f''(\eta) \end{aligned}$$

2. Правые прямоугольники

$$\varepsilon_{\text{прав. пр.}} = -\frac{(b-a)^2}{2N} f'(\eta)$$

3. Средние прямоугольники

$$\begin{aligned} \varepsilon_k &= \frac{(x_{k+1} - x_k)^3}{24} f''(\eta_k) = \frac{(b-a)^3}{24N^3} f''(\eta_k) \\ \varepsilon_{\text{сред. пр.}} &= \sum_{k=1}^N \varepsilon_k = \frac{(b-a)^3}{24N^2} f''(\eta) \end{aligned}$$

4. Трапеции

$$\begin{aligned} \varepsilon_k &= -\frac{(x_{k+1} - x_k)^3}{12} f''(\eta_k) = -\frac{(b-a)^3}{12N^3} f''(\eta_k) \\ \varepsilon_{\text{трап}} &= \sum_{k=1}^N \varepsilon_k = -\frac{(b-a)^3}{12N^2} f''(\eta) \end{aligned}$$

5. Формула Симпсона

Тут следует учитывать, что длина каждого участка в два раза больше, чем ранее, а число таких участков в два раза меньше – $N/2$.

$$\begin{aligned} \varepsilon_k &= -\frac{(x_{k+2} - x_k)^5}{2880} f^{(4)}(\eta_k) = -\frac{(b-a)^5}{90N^5} f^{(4)}(\eta_k) \\ \varepsilon_{\text{Симпс}} &= \sum_{k=1}^{N/2} \varepsilon_k = -\frac{(b-a)^5}{180N^4} \left[\frac{1}{N/2} \sum_{k=1}^{N/2} f^{(4)}(\eta_k) \right] = -\frac{(b-a)^5}{180N^4} f^{(4)}(\eta) \end{aligned}$$

Заметим, что все эти формулы описываются общей зависимостью

$$\varepsilon = \alpha \frac{(b-a)^{p+1}}{N^p} f^{(p)}(\eta) \quad (44)$$

Вопрос 10. Общий подход к построению квадратурных формул. Квадратурные формулы Ньютона-Котеса, Чебышева, Гаусса.

10.1 Общий подход к построению квадратурных формул.

Заметим, что все полученные выше простейшие квадратурные формулы имеют следующий вид

$$\int_a^b f(x)dx \approx \sum_{k=1}^S A_k f(x_k) \quad (45)$$

Узлы x_k и веса A_k квадратурной формулы получались на основе интегрирования соответствующих интерполяционных полиномов. Поставим задачу несколько иначе. Требуется выбрать узлы и веса так, чтобы формула (45) была бы *точной* для полиномов заданной степени. Логика таких требований очевидна. Если подынтегральная функция хорошо аппроксимируется этим полиномом, то и формула (45) обеспечит требуемую погрешность решения задачи. В противном случае, промежуток $[a, b]$ всегда можно разбить на достаточно малые промежутки и применить составные квадратурные формулы.

Потребуем, чтобы формула (45) была бы точна для полинома нулевой степени $f(x) = \alpha = \text{const}$. Вынося константу α из-под знаков интеграла и суммы и сокращая на нее, имеем

$$\sum_{k=1}^S A_k = b - a$$

Второе уравнение получим, требуя точности для полинома первой степени и поставив с этой целью $f(x) = x$.

$$\sum_{k=1}^S A_k x_k = \frac{b^2 - a^2}{2}$$

Это же требование для $f(x) = x^2$ выглядит следующим образом

$$\sum_{k=1}^S A_k x_k^2 = \frac{b^3 - a^3}{3}$$

а в общем случае для $f(x) = x^N$ условие с номером $N + 1$ имеет вид

$$\sum_{k=1}^S A_k x_k^N = \frac{b^{N+1} - a^{N+1}}{N + 1}$$

Объединяя все уравнения, получим систему из $N + 1$ уравнения относительно $2S$ неизвестных x_k и A_k . Эта система является общей для многих семейств квадратурных формул, отличающихся друг от друга дополнительными условиями, накладываемыми на x_k и A_k .

10.2 Квадратурные формулы Ньютона-Котеса

Узлы квадратурной формулы здесь выбираются равноотстоящими

$$h = \frac{b-a}{S-1}, \quad x_k = a + (k-1)h, \quad x_1 = a \quad x_S = b$$

В данном случае система является линейной относительно S относительно A_k и легко решается. Ее определитель является определителем Вандермонда, что обеспечивает единственность решения. Для составных квадратурных формул при удвоении N (числа внутренних промежутков) в половине возникающих узлов x_k значения функции $f(x_k)$ уже вычислялись ранее и могли быть сохранены, что позволяет сократить объем вычислений вдвое.

Как результат, имеем систему из S уравнений с S неизвестными A_k , и эти квадратурные формулы оказываются гарантированно точными для полиномов степени $N = S - 1$.

Заметим, что, решая систему последовательно для $S = 1, 2, 3$, можно прийти к уже знакомым формулам прямоугольников, трапеций и Симпсона.

10.3 Квадратурные формулы Чебышева

Когда значения $f(x_k)$ определены с заметной погрешностью, удобно установить равные веса и решать задачу только относительно x_k .

$$\int_a^b f(x)dx \approx A \sum_{k=1}^S f(x_k)$$

Получаем систему из $S + 1$ уравнения относительно такого же числа неизвестных x_k (S) и A (1), что позволяет в случае успешного решения получить формулы, гарантированно точные для полиномов степени $N = S$. Однако, учитывая нелинейность системы, вопросы существования и единственности выходят на первый план. Оказывается, что система решается единственным образом для $S \in [1; 7] \cup \{9\}$. С.Н.Берштейном было показано, что для других значений S формулы Чебышева не существуют.

10.4 Квадратурные формулы Гаусса

В этих формулах на узлы и веса не накладываются никакие дополнительные условия, и все свободные $2S$ параметров используются при решении системы из $2S$ уравнений. В отличие от формул Чебышева, формулы Гаусса существуют для любого числа узлов. Они гарантированно точны для полиномов степени $N = 2S - 1$ и называются *формулами наивысшей алгебраической степени точности*.

На практике для получения A_k и x_k нет никакой необходимости обращаться к системе. Результаты ее решения для промежутков $[-1, 1]$ и $[0, 1]$ приведены в различных справочниках, а пользователю достаточно лишь воспользоваться заменой переменных в (45). Так для промежутка $[-1, 1]$ эта замена выглядит следующим образом: $x = \frac{a+b}{2} + \frac{b-a}{2}t$. При изменении значений t от -1 до 1 , переменная x пробегает

значения от a до b . Учитывая $dx = \frac{b-a}{2}dt$, формула (45) имеет вид

$$\int_a^b f(x)dx = \frac{b-a}{2} \int_{-1}^1 f\left(\frac{a+b}{2} + \frac{b-a}{2}t\right) dt \approx \frac{b-a}{2} \sum_{k=1}^S A_k f\left(\frac{a+b}{2} + \frac{b-a}{2}t_k\right) \quad (46)$$

где веса A_k и узлы t_k берутся из справочника.

Вопрос 11. Адаптивные квадратурные формулы.

Подпрограмма QUANC8.

Рассмотренный ранее простой алгоритм вычисления интеграла на основе составных квадратурных формул со сравнением результата для N и $2N$, отличается достаточной надежностью, но его быстродействие может быть заметно повышено. Например, если функция в основном ведет себя стабильно, а на каком-то малом участке резко меняется, то имеет смысл требовать подсчет с малым шагом только на этом участке, а не на всем рассматриваемом промежутке.

Целесообразным представляется построение алгоритма, который был бы способен *адаптироваться* к виду функции и выбирать достаточно малый шаг там, где функция меняется быстро и характеризуется большими производными, и относительно большой шаг там, где функция меняется медленно. На этом пути возможны два варианта: минимизировать погрешность при заданном объеме вычислений или минимизировать объем вычислений при заданных требованиях к погрешности. В рассматриваемой программе реализован второй подход.

В основу положена квадратурная формула Ньютона-Котеса с девятью узлами, т.е. восемью промежутками между ними, что и оправдывает название программы. Ее составная формула имеет погрешность вида (32) для $p = 10$.

Рассмотрим промежуток длиной h_k внутри $[a, b]$ и введем для него следующие обозначения:

I_k – точное значение интеграла на этом промежутке

P_k – значение интеграла, вычисленное по квадратурной формуле с девятью узлами

Q_k – значение интеграла, вычисленное по той же формуле, примененной к двум половинам этого промежутка (по сути используется составная формула с вдвоем большим значением N)

Учитывая вид (32) для $p = 10$, для погрешностей P_k и Q_k последовательно имеем

$$I_k - P_k \approx 2^p (I_k - Q_k); \quad I_k \approx \frac{2^p Q_k - P_k}{2^p - 1}; \quad I_k - Q_k \approx \frac{Q_k - P_k}{2^p - 1} = \frac{Q_k - P_k}{1023}$$

Обозначая требуемую абсолютную погрешность вычисления интеграла на всем промежутке $[a, b]$ за ε_A , считаем промежуток h_k «принятым», а интеграл на нем вычисленным, если выполняется неравенство

$$\left| \frac{Q_k - P_k}{1023} \right| \leq \frac{h_k}{b-a} \varepsilon_A \quad (47)$$

Множитель $\frac{h_k}{b-a}$ является весовым коэффициентом и отражает вклад погрешности на промежутке h_k в общую погрешность для всего промежутка. Возможно также использование и относительной погрешности ε_R

$$\left| \frac{Q_k - P_k}{1023} \right| \leq \frac{h_k}{b-a} \varepsilon_R |\tilde{I}_k| \quad (48)$$

где \tilde{I}_k – оценка вычисления интеграла по всему промежутку. Следует, однако, помнить, что использование критерия относительной погрешности усложняется, если значение \tilde{I}_k оказывается нулевым или близким к нулю. В программе QUANC8 пользователю предоставляется возможность использовать один из трех вариантов контроля погрешности на основе формул (47) и (48)

$$\left| \frac{Q_k - P_k}{1023} \right| \leq \frac{h_k}{b-a} \max(\varepsilon_A; \varepsilon_R |\tilde{I}_k|) \quad (49)$$

1. $\varepsilon_R = 0$, $\varepsilon_A \neq 0$ – контроль абсолютной погрешности
2. $\varepsilon_R \neq 0$, $\varepsilon_A = 0$ – контроль относительной погрешности
3. $\varepsilon_R \neq 0$, $\varepsilon_A \neq 0$ – контроль «смешанной» погрешности

В последнем случае делается попытка избежать упомянутых неприятных ситуаций со значениями \tilde{I}_k , близкими к нулю.

Адаптация программы к виду функции и реализация переменного шага интегрирования реализуются в соответствии со следующим алгоритмом. Вычисляются P_k и Q_k применительно ко всему промежутку. Если погрешность еще достаточно велика, промежуток делится пополам, значения подынтегральной функции $f(x)$, вычисленные на правой половине промежутка, запоминаются, и все повторяется для левой половины промежутка. Такое обращение каждый раз к левой половине текущего промежутка продолжается до тех пор, пока крайний слева промежуток не будет принят. После этого обрабатывается ближайший к нему правый промежуток. Запоминание значений $f(x)$ повышает быстроедействие алгоритма.

В программе реализовано два ограничения сверху на объем вычислений. Во-первых, деление промежутка пополам продолжается не более 30 раз. По достижении этой величины соответствующий интеграл на нем считается вычисленным, а промежуток «принятым», независимо от условия (49). Число таких промежутков, принятых с нарушением условия (49), содержится в целой части выходного значения переменной FLAG. Длина каждого такого промежутка крайне мала ($\approx 10^{-9}$), и подобная ситуация, как правило, связана с разрывами подынтегральной функции или ее «зашумлением» вычислительной погрешностью. Во-вторых, вводится ограничение сверху на количество вычислений подынтегральной функции $f(x)$. Если этот предел достигнут, то информация о точке x^* , где возникла трудность, отражена в дробной части выходного значения переменной FLAG.

Например, для FLAG = 91.21, число промежутков равно 91, а

$$\frac{b - x^*}{b - a} = 0.21; \quad x^* = b - 0.21(b - a)$$

Программа имеет следующие параметры

FUN – имя подпрограммы-функции, вычисляющей значение подынтегральной функции $f(x)$;

A, **B** – нижний и верхний пределы интегрирования;

ABSERR, **RELERR** – границы абсолютной ε_A и относительной ε_R погрешностей.

Остальные параметры – выходные со следующим смыслом:

RESULT – значение интеграла, определенное программой;

ERREST – оценка погрешности, выполненная программой и удовлетворяющая (49);

NOFUN – количество вычислений подынтегральной функции $f(x)$, использованных для получения результата;

FLAG – индикатор надежности результата. Нулевое значение этой переменной отвечает относительной надежности результата, а ненулевое, как уже отмечалось, свидетельствует об отклонениях от нормального хода выполнения программы.

Вопрос 12. Задача численного дифференцирования.

Влияние вычислительной погрешности.

12.1 Задача численного дифференцирования.

Предлагаемая задача ставится следующим образом. Для таблично заданной функции $f(x)$ требуется оценить значения производной функции в узлах таблицы.

Идея, лежащая в основе численного дифференцирования, крайне проста и уже использовалась при получении квадратурных формул. Исходная функция аппроксимируется интерполяционным полиномом

$$f(x) = Q_m(x) + R_m(x)$$

и производная от полинома дает формулу численного дифференцирования

$$\frac{df(x_k)}{dx} \approx \frac{dQ_m(x_k)}{dx}$$

а производная от остаточного члена позволяет оценить погрешность этой операции

$$\varepsilon = \frac{dR_m(x_k)}{dx}$$

Ограничимся случаем, когда узлы таблицы будут равноотстоящими с шагом $h = x_{k+1} - x_k$. Начнем с полинома первой степени, построенного по двум узлам x_k и x_{k+1}

$$Q_1(x) = \frac{x - x_{k+1}}{x_k - x_{k+1}} f(x_k) + \frac{x - x_k}{x_{k+1} - x_k} f(x_{k+1})$$

Дифференцируя его и полагая последовательно $x = x_k$ и $x = x_{k+1}$, получаем

$$\frac{df(x_k)}{dx} \approx \frac{f_{k+1} - f_k}{h} \tag{50}$$

$$\frac{df(x_{k+1})}{dx} \approx \frac{f_{k+1} - f_k}{h} \tag{51}$$

Хотя правые части обоих выражений равны, формулы получились принципиально различными. Выполним аналогичные операции для полинома второй степени

$$Q_2(x) = \frac{(x - x_{k+1})(x - x_{k+2})}{(x_k - x_{k+1})(x_k - x_{k+2})}f(x_k) + \frac{(x - x_k)(x - x_{k+2})}{(x_{k+1} - x_k)(x_{k+1} - x_{k+2})}f(x_{k+1}) + \\ + \frac{(x - x_k)(x - x_{k+1})}{(x_{k+2} - x_k)(x_{k+2} - x_{k+1})}f(x_{k+2})$$

Последовательно полагая $x = x_k$, $x = x_{k+1}$ и $x = x_{k+2}$, получаем

$$\frac{df(x_k)}{dx} \approx \frac{-3f_k + 4f_{k+1} - f_{k+2}}{2h} \quad (52)$$

$$\frac{df(x_{k+1})}{dx} \approx \frac{f_{k+2} - f_k}{2h} \quad (53)$$

$$\frac{df(x_{k+2})}{dx} \approx \frac{3f_{k+2} - 4f_{k+1} + f_k}{2h} \quad (54)$$

Для оценки погрешности всех формул необходимо продифференцировать остаточный член $R_m(x)$. Для полинома первой степени он имеет вид

$$R_1(x) = \frac{(x - x_k)(x - x_{k+1})}{2!}f''(\eta)$$

$$\frac{dR_1(x)}{dx} = \frac{x - x_{k+1}}{2!}f''(\eta) + \frac{x - x_k}{2!}f''(\eta) + \frac{(x - x_k)(x - x_{k+1})}{2!}f'''(\eta)\eta'(x)$$

Спасает ситуацию то, что оценивать погрешность нужно в узлах интерполирования. Так как при $x = x_k$ и $x = x_{k+1}$ два слагаемых из трех в этой формуле обращаются в ноль

$$\varepsilon_1(x_k) = \frac{dR_1(x_k)}{dx} = -\frac{h}{2}f''(\eta) \quad (55)$$

$$\varepsilon_1(x_{k+1}) = \frac{dR_1(x_{k+1})}{dx} = \frac{h}{2}f''(\eta) \quad (56)$$

Эти два выражения задают погрешность численного дифференцирования для формул (50) и (51) соответственно. Аналогично продифференцируем погрешность $R_2(x)$

$$R_2(x) = \frac{(x - x_k)(x - x_{k+1})(x - x_{k+2})}{3!}f'''(\eta)$$

Последовательно подставляя в результат $x = x_k$, $x = x_{k+1}$ и $x = x_{k+2}$, получаем

$$\varepsilon_2(x_k) = \frac{dR_2(x_k)}{dx} = \frac{h^2}{3}f'''(\eta) \quad (57)$$

$$\varepsilon_2(x_{k+1}) = \frac{dR_2(x_{k+1})}{dx} = -\frac{h^2}{6}f'''(\eta) \quad (58)$$

$$\varepsilon_2(x_{k+2}) = \frac{dR_2(x_{k+2})}{dx} = \frac{h^2}{3}f'''(\eta) \quad (59)$$

Заметим, что на меньшую погрешность можно рассчитывать, используя формулу (53), которая и является наиболее популярной на практике. Формулы (52) и (54) используются для дифференцирования в начале и в конце таблицы соответственно.

Если интерполяционный полином второй степени продифференцировать дважды, то получается простейшая формула для второй производной

$$\frac{d^2 f(x_{k+1})}{dx^2} \approx \frac{f_{k+2} - 2f_{k+1} + f_k}{h^2} \quad (60)$$

Практически важным является вопрос о выборе шага h для формул численного дифференцирования. Ограничение сверху накладывается величиной погрешности ε_2 , а снизу – точностью задания табличных данных для $f(x)$.

12.2 Влияние погрешности задания функции на точность.

В качестве примера вновь обратимся к простейшей формуле для первой производной (50). Пусть в ней значение f_{k+1} определено с погрешностью Δ_{k+1} , а значение $f_k - \Delta_k$. Тогда общая погрешность $\varepsilon(h)$ складывается из двух погрешностей

$$\varepsilon(h) = \varepsilon_1(h) + \varepsilon_2(h)$$

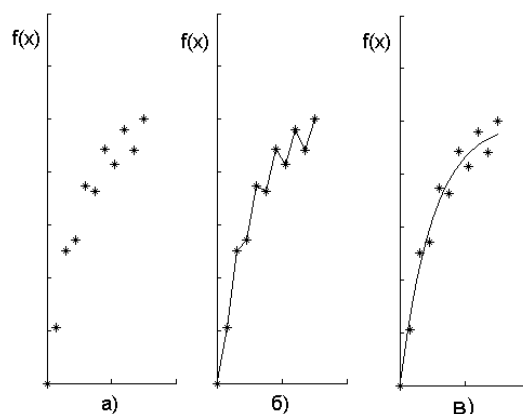
первая из которых задается формулой (56) и примерно линейно убывает с уменьшением шага h , а вторая зависит от Δ_k и Δ_{k+1} . Оценивая полную погрешность сверху, получаем

$$|\varepsilon(h)| \leq |\varepsilon_1(h)| + |\varepsilon_2(h)| = \frac{h}{2} |f''(\eta)| + \frac{|\Delta_{k+1} - \Delta_k|}{h} \quad (61)$$

Оптимальное значение шага h_{opt} отвечает ситуации, когда оба слагаемых равны друг другу. На практике в точном определении h_{opt} нет необходимости, важно лишь знать о характере зависимости (61).

Вопрос 13. Среднеквадратичная аппроксимация (дискретный случай). Понятие веса.

Критерий интерполирования, предполагающий совпадение исходной и аппроксимирующей функции в узлах таблицы, не является единственным. Обратимся к экспериментальным данным, представленным на рисунке (а).



Если выполнить по ним интерполяцию, то получится кривая на рис. (б). Маловероятно, чтобы ее вид отвечал исходной зависимости, положенной в основу таблицы. Вероятнее всего, что этой зависимости отвечает кривая, похожая на рис. (в), а отклонение экспериментальных данных от нее продиктовано сравнительно большой погрешностью измерений. В таком случае целесообразно использовать среднеквадратичный критерий

$$\rho^2 = \sum_{k=1}^N (f(x_k) - g(x_k))^2$$

или, если аппроксимируемая функция задана непрерывно,

$$\rho^2 = \int_a^b (f(x) - g(x))^2 dx$$

Близость функций по среднеквадратичному критерию еще не гарантирует малой величины их максимальной разности

$$\delta = \max_{[a,b]} |f(x) - g(x)|$$

Малое значение интеграла или суммы свидетельствует лишь о том, что почти на всем отрезке $[a, b]$ значения $f(x)$ и $g(x)$ мало отличаются друг от друга, хотя в отдельных точках или на небольших отрезках разность их значений может быть значительной. Проблема выбора аппроксимирующей функции решается так же, как и при интерполяции: $Q(x)$ выбирается в виде обобщенного многочлена:

$$Q_m(x) = \sum_{k=0}^m a_k \varphi_k(x) \quad (62)$$

где $\{\varphi_k\}$ – заданный набор линейно независимых функций, а коэффициенты a_k подлежат определению.

13.1 Дискретный случай. Весовые коэффициенты.

Функция $f(x)$ задается на дискретном множестве точек таблицей, а ее аппроксимация $Q_m(x)$ выбирается в виде обобщенного многочлена (62). Коэффициенты a_k

выбираются из условия минимума величины ρ^2

$$\rho^2 = \sum_{i=1}^N (Q_m(x_i) - f(x_i))^2 \rightarrow \min \quad (63)$$

Рассмотрим три варианта соотношения N и m .

1. $N = m + 1$

Число коэффициентов a_k равно числу точек таблицы, решение задачи единственное, и им является интерполяционный полином, проходящий через все точки. Минимальное значение $\rho^2 = 0$.

2. $N < m + 1$

Минимальное значение ρ^2 также равно нулю, но задача имеет бесконечное множество решений.

3. $N > m + 1$

Это типичный случай среднеквадратичной аппроксимации. Более того, на практике часто $N \gg m + 1$. Минимальное значение ρ^2 оказывается уже, как правило, ненулевым, а задача имеет единственное решение. Рассмотрим этот вариант подробнее.

Записываем необходимое условие экстремума

$$\frac{\partial \rho^2}{\partial a_k} = 0, \quad k = 0, 1, \dots, m$$

и выполняем операцию дифференцирования:

$$\frac{\partial \rho^2}{\partial a_k} = 2 \sum_{i=1}^N (Q_m(x_i) - f(x_i)) \varphi_k(x_i) = 0$$

Подставляя в получившуюся формулу выражение для $Q_m(x)$, получаем систему линейных алгебраических уравнений относительно a_k :

$$a_0 \sum_{i=1}^N \varphi_0(x_i) \cdot \varphi_k(x_i) + a_1 \sum_{i=1}^N \varphi_1(x_i) \cdot \varphi_k(x_i) + \dots + a_m \sum_{i=1}^N \varphi_m(x_i) \cdot \varphi_k(x_i) = \sum_{i=1}^N f(x_i) \cdot \varphi_k(x_i) \quad (64)$$

Если определитель системы (64) не равен нулю, задача имеет единственное решение. Самой популярной является аппроксимация полиномами, когда $\varphi_k(x) = x^k$, а система (64) принимает вид

$$\begin{aligned} \left(\sum_{i=1}^N 1 \right) a_0 + \left(\sum_{i=1}^N x_i \right) a_1 + \dots + \left(\sum_{i=1}^N x_i^m \right) a_m &= \left(\sum_{i=1}^N f(x_i) \right) \\ \left(\sum_{i=1}^N x_i \right) a_0 + \left(\sum_{i=1}^N x_i^2 \right) a_1 + \dots + \left(\sum_{i=1}^N x_i^{m+1} \right) a_m &= \left(\sum_{i=1}^N f(x_i) x_i \right) \\ &\dots \\ \left(\sum_{i=1}^N x_i^m \right) a_0 + \left(\sum_{i=1}^N x_i^{m+1} \right) a_1 + \dots + \left(\sum_{i=1}^N x_i^{2m} \right) a_m &= \left(\sum_{i=1}^N f(x_i) x_i^m \right) \end{aligned} \quad (65)$$

При выполнении среднеквадратичной аппроксимации (другое название – «метод наименьших квадратов») возможна ситуация, когда исходные данные имеют различную точность. Если к каким-либо экспериментальным значениям доверие выше, т.е. они являются более надежными по сравнению с другими, это может быть учтено введением в критерий (63) положительных весовых коэффициентов p_i .

$$\rho^2 = \sum_{i=1}^N p_i (Q_m(x_i) - f(x_i))^2 \rightarrow \min \quad (66)$$

Для тех точек, степень доверия к которым выше и к которым аппроксимирующую кривую желательно провести ближе, чем к другим точкам, весовые коэффициенты следует задавать больше. Величину ρ^2 можно трактовать, как своеобразную «функцию штрафа». За отклонение $Q_m(x)$ от $f(x)$ в точке x_i к значению ρ^2 добавляется слагаемое («штраф») $(Q(x_i) - f(x_i))^2$ тем большее, чем больше это отклонение. Если какая-то точка является более приоритетной и к ней аппроксимирующую кривую желательно провести ближе, с помощью весового коэффициента за отклонение в этой точке «штраф» должен быть увеличен.

На практике положительные весовые коэффициенты p_i часто задают так, чтобы их сумма была равна, например, единице или 100. Последнее часто удобно, но не обязательно. Если все коэффициенты умножить на одно и то же число, то, хотя, ρ^2 и изменится, решение задачи останется прежним. Важным являются отношения p_i друг к другу.

Вопрос 14. Среднеквадратичная аппроксимация (непрерывный случай). Понятие ортогональности.

Вопрос 15. Ортогонализация по Шмидту. Примеры ортогональных полиномов.

Вопрос 16. Обратная матрица, собственные числа и

векторы. Задачи на матрицы. Норма матрицы, сходимость матричного степенного ряда, функции от матрицы.

Вопрос 17. 7 теорем о матричных функциях.

Вопрос 18. Решение систем линейных дифференциальных и разностных уравнений с постоянной матрицей.

Вопрос 19. Устойчивость решений дифференциальных и разностных уравнений.

Вопрос 20. Метод Гаусса и явление плохой обусловленности. LU-разложение матрицы. Подпрограммы DECOMP и SOLVE.

Вопрос 21. Метод последовательных приближений для решения линейных систем.

Вопрос 22. Методы бисекции, секущих, обратной

параболической интерполяции для решения нелинейных уравнений. Подпрограмма ZEROIN.

Вопрос 23. Методы последовательных приближений и Ньютона для решения нелинейных уравнений и систем.

Вопрос 24. Задача Коши решения обыкновенных дифференциальных уравнений. Явный и неявный методы ломаных Эйлера, метод трапеций.

Вопрос 25. Методы Адамса. Локальная и глобальная погрешности, степень метода.

Вопрос 26. Методы Рунге-Кутты. Подпрограмма RKF45.

Вопрос 27. Глобальная погрешность. Устойчивость метода. Ограничение на шаг. Явление жесткости и методы решения жестких систем.

Вопрос 28. Метод Ньютона в неявных алгоритмах

решения дифференциальных уравнений.

Вопрос 29. Сведение дифференциального уравнения
высокого порядка к системе уравнений первого
порядка. Метод стрельбы для решения краевых
задач.
