

П.4 Тригонометрическая интерполяция.

1. В тригонометрической интерполяции в отличие от других видов интерполяции, интерполяция происходит не по x_0, x_1, \dots, x_n , а по x_0, x_1, \dots, x_{n-1} точкам, т.к.

интерполирование происходит периодически и $y_0 = y_n$, т.е. период $T = x_n - x_0$

2. $x_i = x_0 + ih$, $i = \overline{0, n-1}$ - равноотстоящие узлы интерполяции.

4.1. Формулы Т.И.:

$$y(x) = \sum_{-\frac{n}{2} < j \leq \frac{n}{2}} A_j \exp(2\pi j \frac{x-x_0}{nh}) \quad (4.18a)$$

$$\text{где } nh = T, \quad A_j = \sum_{k=0}^{n-1} y_k \exp(-2\pi i \frac{kj}{n}) \quad (4.18б)$$

Замечание:

в этих формулах i – мнимая единица и для работы по (4.18a), (4.18б) нужна формула

Эйлера: $e^{ix} = \cos x + i \sin x$

При Т.И. интерполирующая функция $y(x)$:

1) периодична с периодом $nh = T$.

2) в узлах интерполяции $y(x_i) = y_i$, т.е. если $y(x_i)$ – вещественное, то в узлах мнимая часть y – нулевая.

3) в промежуточных точках y может принимать комплексные значения, но $\text{Im } y$ – будет не велика и её можно отбросить.

4) если число узлов интерполяции нечётное, т.е. $n=2m+1$, и все y_i – вещественные, то функция y полученная по (4.18a), (4.18б) сама по себе будет вещественна.

Коэффициенты A_j – комплексные, а y_i – вещественные функции и в этом случае

вычисления можно осуществлять не с комплексными, а с вещественными числами по формуле (4.19).

$$(4.19a) \quad y(x) = b_0 + \sum_{j=1}^m b_j \cos(2\pi j \frac{x-x_0}{nh}) + \sum_{j=1}^m a_j \sin(2\pi j \frac{x-x_0}{nh})$$

$$(4.19б) \quad b_0 = \frac{1}{n} \sum_{k=0}^{n-1} y_k; \quad b_j = \frac{1}{n} \sum_{k=0}^{n-1} y_k \cos(2\pi \frac{kj}{n}); \quad a_j = \frac{1}{n} \sum_{k=0}^{n-1} y_k \sin(2\pi \frac{kj}{n}) \quad j = \overline{0, n-1}$$

4.2. Быстрое преобразование Фурье.

Определение: преобразование набора значений функции ($y_0 \dots y_{n-1}$) в набор коэффициентов ($A_0 \dots A_{n-1}$) (используя (4.18б)), участвующих в разложении Фурье, называется прямым преобразованием Фурье (ППФ), а обратным преобразованием Фурье (ОПФ) – преобразование массива A_j в y_k (по (4.18б)).

Если осуществлять эти вычисления непосредственно по (4.18a, б), то трудоёмкость – $\tau = n^2$ (т.к. имеем n коэффициентов, в каждом из которых n слагаемых).

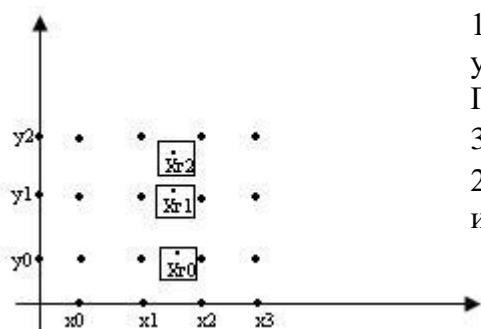
Эти же вычисления можно делать по более быстрым формулам – быстрое преобразование Фурье. Трудоёмкость по этим формулам существенно меньше: не n^2 , а $n \log_2 n$.

4.3. Многомерная интерполяция.

Пусть мы имеем функцию нескольких переменных, значения которой нам известны в некоторых точках (при задаче интерполирования нам надо знать значение функции f в наперед заданной точке).

Решим простой вариант двумерной интерполяции $f(x, y)$:

(x_k, y_l) - узлы образующие прямоугольную сетку.



1) Интерполируем функцию по x (при фиксированном y) и получим значение функции в точке x .

При фиксированном x , 1 раз интерполируем по y (по 3-м x_i) и получим значение в точке x_{r2} .

2) x и y можно поменять местами и сделать интерполяцию по y и 1 раз по x .

Значения полученные этими способами весьма близки к точным значениям функции, близки друг к другу, но могут и различаться.

П.5. Применение интерполяции.

5.1. Обратная интерполяция.

С помощью обратной интерполяции можно решать нелинейные уравнения.

Решим $f(x)=a$:

Идея обратной интерполяции: пусть f вблизи корня уравнения $f(x)=a$ – монотонно возрастает или убывает, тогда у неё существует обратная функция.

$g(x)$ – обратная функция, значение которой в точке a нас не интересует.

$f(x)=a$; $g = f^{-1}$

и будет искомым корнем уравнения $f(x)=a$, $f^{-1}(a) = X$

Возьмём интервал $[x_0, x_n]$, на котором f – монотонна и имеет обратную функцию, следовательно, мы знаем $y_i = f(x_i)$

Применим интерполяцию для вычисления значений обратной функции g и найдем значение интерполирующей функции в точке a . Это и будет, приблизительно, искомым корень.

$x=g(a)$

При этом при интерполяции x и y меняются местами, так как мы интерполируем не f а g .

Пример:

x	10	15	17	20
y	3	7	11	17

$f(x)=10$

Решение находим интерполируя обратную функцию по 4-м точкам (например, по формуле Лагранжа).

$$g(10) = x_0 \frac{(y - y_1)(y - y_2)(y - y_3)}{(y_0 - y_1)(y_0 - y_2)(y_0 - y_3)} + x_1 \frac{(y - y_0)(y - y_2)(y - y_3)}{(y_1 - y_0)(y_1 - y_2)(y_1 - y_3)} +$$

$$+ x_2 \frac{(y - y_0)(y - y_1)(y - y_3)}{(y_2 - y_0)(y_2 - y_1)(y_2 - y_3)} + x_3 \frac{(y - y_0)(y - y_1)(y - y_2)}{(y_3 - y_0)(y_3 - y_1)(y_3 - y_2)} = 10 \frac{(10 - 7)(10 - 11)(10 - 17)}{(3 - 7)(3 - 11)(3 - 17)} +$$

$$+ 15 \frac{(10 - 3)(10 - 11)(10 - 17)}{(7 - 3)(7 - 11)(7 - 17)} + 17 \frac{(10 - 3)(10 - 7)(10 - 17)}{(11 - 3)(11 - 7)(11 - 17)} + 20 \frac{(10 - 3)(10 - 7)(10 - 11)}{(17 - 3)(17 - 7)(17 - 11)} = 16,64$$

5.2. Численное дифференцирование функции.

5.2.1. Постановка задачи численного дифференцирования.

В точках x_i , $i = \overline{0, n}$ - известны значения функции $y_i = f(x_i)$. Задача численного дифференцирования – найти значение производной f' или f'' или ... $f^{(n)}$ в любой наперед заданной точке x . Поступаем также как при интерполяции.

Общие идеи:

Заменяем неизвестную функцию f на интерполирующий многочлен P .

$f' \rightarrow P'$; $f'' \rightarrow P''$, а её производная И.М. Продифференцировав формулы для И.М. Схема

Эйткена сразу отпадает, т.к. это схема, а не формула.

Формула Лагранжа громоздка, следовательно, будем дифференцировать формулу Ньютона И.М.

5.2.2. Формулы численного дифференцирования.

Рассмотрим 1-ую формулу Ньютона И.М.:

$$y(x) = y_0 + \frac{\Delta y_0}{1!} q + \frac{\Delta^2 y_0}{2!} q(q-1) + \frac{\Delta^3 y_0}{3!} q(q-1)(q-2) + \frac{\Delta^4 y_0}{4!} q(q-1)(q-2)(q-3) + \dots + \frac{\Delta^n y_0}{n!} q(q-1)(q-2)\dots(q-n), \quad \text{где} \quad q = \frac{x - x_0}{h}$$

дифференцируем по x :

$$\frac{dy}{dx} = \frac{dy}{dq} * \frac{dq}{dx} = \frac{1}{h} (0 + \Delta y_0 + \frac{\Delta^2 y_0}{2} (2q-1) + \frac{\Delta^3 y_0}{6} (3q^2 - 6q + 2) + \frac{\Delta^4 y_0}{12} (2q^3 - 9q^2 + 11q - 3) + \dots) \approx y'(x), \quad (5.1)$$

Формулу в (5.1) дифференцируем по y :

$$y''(x) = \frac{1}{h^2} (0 + 0 + \Delta^2 y_0 + (q-1)\Delta^3 y_0 + \frac{\Delta^4 y_0}{12} (6q^2 - 18q + 11) + \dots) \quad [5.2]$$

В формулах (5.1) и (5.2) решение можно обрывать раньше. При этом, если в этих формулах до k , то мы получим производную И.М., которая интерполирует функцию не во всех (x_0, x_1, \dots, x_n) , а только (x_0, x_1, \dots, x_k) точках.

Пусть в (5.1) и (5.2) $x = x_0$, т.е. $q=0$, получаем:

$$(5.3) \quad y'(x_0) = \frac{1}{h} (\Delta y_0 - \frac{1}{2} \Delta^2 y_0 + \frac{1}{3} \Delta^3 y_0 - \frac{1}{4} \Delta^4 y_0 + \dots)$$

$$(5.4) \quad y''(x_0) = \frac{1}{h^2} (\Delta^2 y_0 - \Delta^3 y_0 + \frac{11}{12} \Delta^4 y_0 - \dots)$$

На практике удобнее дифференцировать не односторонние формулы (1,2 формулы Ньютона), а центральные (формулу Стирлинга), так как узлы интерполяции располагаются симметрично относительно начальной точки x_0 . Возьмём в формуле Стирлинга первые три слагаемых (интерполяция по трём точкам x_{-1}, x_0, x_1), получим:

$$p_2(x) = y_0 + \frac{\Delta y_{-1} + \Delta y_0}{2} q + \frac{\Delta^2 y_{-1}}{2} q^2$$

$$p'_2(x) = \frac{1}{h} (0 + \frac{\Delta y_{-1} + \Delta y_0}{2} + q \Delta^2 y_{-1})$$

подставим $x = x_0$, т.е. $q = 0$ получим:

$$y'(x) = \frac{1}{h} (\frac{\Delta y_{-1} + \Delta y_0}{2}) = \frac{1}{h} (\frac{(y_0 - y_{-1}) + (y_1 - y_0)}{2}) = \frac{y_1 - y_{-1}}{2h} \quad (5.5)$$

Если же в формуле Стирлинга взять не 3, а 5 первых слагаемых (интерполяция по 5-ти точкам $x_{-2}, x_{-1}, x_0, x_1, x_2$) продифференцируем и подставим $x = x_0$, то получим:

$$y'(x_0) = \frac{1}{12h} (y_{-2} - 8y_{-1} + 8y_1 - y_2) \quad (5.6)$$

Если взять 3 первых слагаемых и продифференцировать дважды по q , то получим:

$$y''(x_0) = \frac{1}{h^2} \Delta^2 y_{-1} = \left(\frac{y_{-1} - 2y_0 + y_1}{h^2} \right) \quad (5.7)$$

5.2.3 Оценка погрешностей численного дифференцирования.

Также как и при интерполяции в численном дифференцировании возникают две погрешности: $\varepsilon_{\text{усеч}}$ и $\varepsilon_{\text{окр}}$.

Погрешность усечения – из-за замены функции на ее интерполирующий многочлен и ее производной на производную от интерполяционного многочлена.

Погрешность округления – из-за того, что значение функции в узлах x_i известны не точно, а с некоторой погрешностью η . Оценим погрешность усечения.

Теорема 5.1:

Погрешность усечения в формуле (5.3) численного дифференцирования (при суммировании k -слагаемых) имеет следующую оценку:

$$\varepsilon_{\text{усеч}} = (-1)^k \frac{h^k}{k+1} f^{(k+1)}(C)$$

где $C \in [x_0, x_k]$.

Доказательство:

$$\begin{aligned} \varepsilon_{\text{усеч}} &= f'(x_0) - p'_k(x_0) = (f(x) - p_k(x))'|_{x=x_0} = \left\{ \text{берём формулу из теоремы 4.3} \right\} = \\ &= \frac{1}{(k+1)!} (f^{(k+2)}(C) \cdot C'(x) \cdot (x-x_0)(x-x_1)\dots(x-x_k) + f^{(k+1)}(C) \cdot 1 \cdot (x-x_1)\dots(x-x_k) + \\ &+ f^{(k+1)}(C) \cdot (x-x_0) \cdot 1 \cdot (x-x_2)\dots(x-x_k)) \Big|_{x=x_0} = \\ &= \frac{1}{(k+1)!} (0 + f^{(k+1)}(C) \cdot 1 \cdot (x_0-x_1)(x_0-x_2)\dots(x_0-x_k) + 0 + \dots + 0) = \{x-x_0 = qh\} = \\ &= \frac{1}{(k+1)!} f^{(k+1)}(C)(-h)(-2h)(-3h)\dots(-kh) = \frac{(-1)^k f^{(k+1)}(C)}{k+1} h^k \quad (5.8) \end{aligned}$$

Замечания:

При доказательстве теоремы был использован тот факт, что $C=C(x)$ и $C'(x)$ – существует. Это будет так, если функция f была достаточно гладкой.

Из-за того, что $C'(x)$ мы вообще никак не можем оценить, погрешность усечения мы можем находить только в узлах интерполяции, с тем, чтобы 1-ое слагаемое, где присутствует $C'(x)$, занулилось.

На практике формулу (5.8) мы заменяем на формулу (5.9) (оценка сверху для $\varepsilon_{\text{усеч}}$)

$$\varepsilon_{\text{усеч}} = \frac{M_{k+1}}{(k+1)} h^k \quad (5.9)$$

где $M_{k+1} = \max_{x \in (x_0, x_k)} |f^{(k+1)}(x)|$

Вспомним, что конечная разность очень похожа на производную ($\Delta^k y \approx y^k h^k$).

Тогда (5.9) можно заменить на (5.10):

$$\varepsilon_{\text{усеч}} \approx \frac{\Delta^{(k+1)} y_0}{(k+1)h}$$

Для формул (5.5), (5.6) и (5.7) $\varepsilon_{\text{усеч}}$ можно вывести таким же образом, как и в теореме 5.1, получаем:

$$\text{Для (5.5)} \rightarrow \varepsilon_{\text{усеч}} \leq \frac{1}{6} h^2 M_3$$

$$\text{Для (5.6)} \rightarrow \varepsilon_{\text{усеч}} \leq \frac{1}{30} h^4 M_5$$

$$\text{Для (5.7)} \rightarrow \varepsilon_{\text{усеч}} \leq \frac{1}{12} h^2 M_4$$

$$\text{где } M_3 = \max_{x \in [x_{-1}, x_1]} |f^{(3)}(x)|, \quad M_5 = \max_{x \in [x_{-2}, x_2]} |f^{(5)}(x)|, \quad M_4 = \max_{x \in [x_{-1}, x_1]} |f^{(4)}(x)|$$

Оценим $\varepsilon_{\text{окр}}$ для центральных формул.

Рассмотрим формулу (5.5)

$$y'(x) = \frac{y_1 - y_{-1}}{2h}, \quad y_1 \leq \eta, y_{-1} \leq \eta \Rightarrow (y_1 - y_{-1}) \leq 2\eta \text{ таким образом } \varepsilon_{\text{окр}} : \frac{y_1 - y_{-1}}{2h} \leq \frac{2\eta}{2h} = \frac{\eta}{h}$$

Аналогично:

$$\text{для (5.6)} \rightarrow \varepsilon_{\text{окр}} \leq \frac{18\eta}{2h} = \frac{9\eta}{h}$$

$$\text{для (5.7)} \rightarrow \varepsilon_{\text{окр}} \leq \frac{4\eta}{h^2}$$

Заметим, что во всех формулах при $h \rightarrow \infty$, $\varepsilon_{\text{усеч}} \rightarrow 0$, $\varepsilon_{\text{окр}} \rightarrow \infty$ и при

$$h \rightarrow 0, \quad \varepsilon_{\text{усеч}} \rightarrow \infty, \quad \varepsilon_{\text{окр}} \rightarrow 0$$

Поэтому имеем следующую картину:

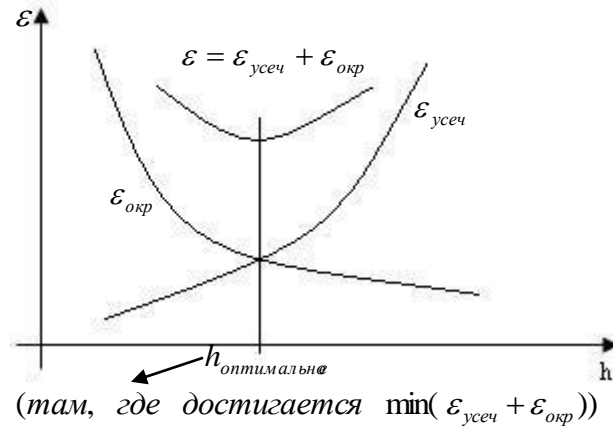


Таблица для погрешностей центральных формул:

	$\varepsilon_{\text{усеченное}}$	$\varepsilon_{\text{округления}}$	$h_{\text{оптимальное}}$	ε_{\min}
4.6	$1/6 h^2 M_3$	η / h	$\sqrt[3]{3\eta / M_3}$	$\sqrt[3]{9\eta^2 M_3 / 2}$
4.7	$1/30 h^4 M_5$	$3\eta / 2h$	$\sqrt[5]{45\eta / 4M_5}$	$15/8 \sqrt[5]{4\eta^4 M_5 / 45}$
4.8	$1/12 h^2 M_4$	$4\eta / h^2$	$\sqrt[4]{48\eta / M_4}$	$2/\sqrt[4]{3} \sqrt[4]{M_4 h}$

П.6. Численное интегрирование.

6.1. Общая идея, решение.

Постановка задачи: в узлах x_i заданы значения функции $y_i = f(x_i)$, $i = \overline{0, n}$. Необходимо

найти значение $\int_a^b f(x)dx$ для любых a, b .

Основная идея численного интегрирования: заменить функцию $f(x)$ на интерполирующую ее функцию, которую мы и будем интегрировать.

$$\int_{x_0}^{x_n} f(x)dx \approx \int_{x_0}^{x_n} p_n(x)dx = \left\{ \text{используя формулу Лагранжа И.М.} \right\} = \int_{x_0}^{x_n} \sum_{i=0}^n y_i q_i(x)dx =$$

$$= \sum_{i=0}^n \int_{x_0}^{x_n} q_i(x)dx = \sum_{i=0}^n y_i A_i, \text{ где } A_i \text{ не зависит от исходной функции } f, \text{ а зависит от узлов интерполяции } x_i.$$

Вычислим A_i :

$$A_i^n = \int_{x_0}^{x_n} q_i(x)dx = \int_{x_0}^{x_n} \frac{(x-x_0)(x-x_1)\dots(x-x_{i-1})(x-x_{i+1})\dots(x-x_n)}{(x_i-x_0)(x_i-x_1)\dots(x_i-x_{i-1})(x_i-x_{i+1})\dots(x_i-x_n)} dx =$$

$$= \left\{ \begin{array}{l} \text{замена} \\ \text{переменных} \\ q = \frac{x-x_0}{h}; dx = dqh \\ x = qh + x_0 \end{array} \right\} = \int_0^n \frac{qh(q-1)h(q-2)h\dots(q-(i-1))h(q-(i+1))h\dots(q-n)h}{ih(i-1)h\dots2h \cdot h(-h)(-2h)\dots(n-i)} h dq =$$

$$= \int_0^n \frac{q(q-1)(q-2)\dots(q-(i-1))(q-(i+1))\dots(q-n)}{i!(n-i)!} (-1)^{n-i} h dq = \frac{(-1)^{n-i} h}{i!(n-i)!} \int_0^n \frac{q(q-1)\dots(q-n)}{q-i} dq =$$

$$= (b-a)H_i^n, \quad \text{где} \quad H_i^n = \frac{(-1)^{n-i}}{i!(n-i)!n} \int_0^n \frac{q(q-1)\dots(q-n)}{q-i} dq \quad (5.11a)$$

Замечания:

Замена A_i^n на H_i^n была сделана с той целью, чтобы коэффициенты Ч.И. не зависели от h , а зависели от n и i .

6.2. Частные случаи, формулы Ньютона - Котеса.

Итак, формула Ч.И. принимает следующий вид:

$$\int_{x_0}^{x_n} f(x)dx \approx (b-a) \sum_{i=0}^n y_i H_i^n \quad (5.11б)$$

где H_i^n вычисляется по формуле (5.11а).

Выпишем частные случаи (5.11а):

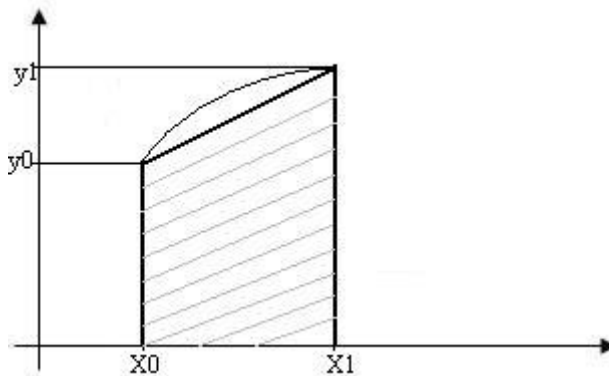
$n=1$

$$H_{0_1}^1 = \frac{(-1)^{1-0}}{0!(1-0)!*1} \int_0^1 \frac{q(q-1)}{q-0} dq = \frac{1}{2}$$

$$H_1^1 = \frac{(-1)^{1-1}}{1!*0!*1} \int_0^1 \frac{q(q-1)}{q-1} dq = \frac{1}{2}$$

Таким образом, при $n=1$, формула Ньютона - Котеса следующий вид:

$\int_{x_0}^{x_1} f(x)dx = (x_1 - x_0)(\frac{1}{2} y_0 + \frac{1}{2} y_1)$ (5.12) – формула трапеций Ч.И. (выражение из правой части площадь трапеции):



Вычислим коэффициенты Н.-К. n=2:

$$H_0^2 = \frac{(-1)^2}{0!(2-0)!2} \int_0^2 \frac{q(q-1)(q-2)}{q-0} dq = \int_0^2 (g^2 - 3g + 2) dg = \frac{1}{4} \left[\frac{g^3}{3} - \frac{3}{4} g^2 + 2g \right] \Big|_0^2 = \frac{1}{6}$$

$$H_1^2 = \frac{(-1)^1}{1!1!2} \int_0^2 \frac{g(g-1)(g-2)}{g-1} dg = -\frac{1}{2} \int_0^2 (g^2 - 2g) dg = -\frac{1}{2} \left[\frac{g^3}{3} - g^2 \right] \Big|_0^2 = \frac{2}{3}$$

$$H_2^2 = \frac{(-1)^0}{2!0!2} \int_0^2 \frac{g(g-1)(g-2)}{g-2} dg = \frac{1}{4} \int_0^2 (g^2 - g) dg = \frac{1}{4} \left[\frac{g^3}{3} - \frac{g^2}{2} \right] \Big|_0^2 = \frac{1}{6}$$

Итак, при n=2, формула Ч.И. принимает следующий вид (формула Симпсона):

$$\int_{x_0}^{x_2} f(x)dx = (x_2 - x_0) \left(\frac{1}{6} y_0 + \frac{2}{3} y_1 + \frac{1}{6} y_2 \right)$$

Аналогичным образом вычисляем коэффициенты при большем n.

Таблица коэффициентов Ньютона - Котеса:

i \ n	1	2	3	4	5	6	7	8
H ₀	1/2	1/6	1/8	7/90	19/288	41/840	751/17280	989/28350
H ₁	1/2	2/3	3/8	32/90	75/288	216/840	3577/17280	5888/28350
H ₂		1/6	3/8	12/90	60/288	27/840	1323/17280	-928/28350
H ₃			1/8	32/90	50/288	272/840	2989/17280	10496/28350
H ₄				7/90	75/288	27/840	2989/17280	*/28350
H ₅					19/288	216/840	1323/17280	10496/28350
H ₆						41/840	3577/17280	-928/28350
H ₇							751/17280	*5888/28350
H ₈								989/28350

6.3. Погрешности формул численного интегрирования.

При численном интегрировании возникают два типа погрешностей: $\varepsilon_{\text{усеч}}$ и $\varepsilon_{\text{окр}}$

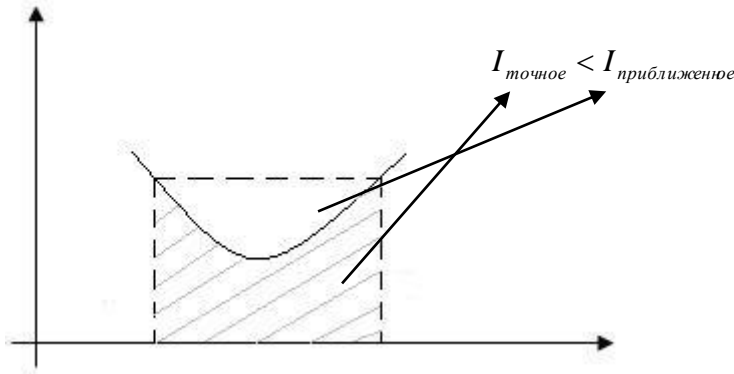
Погрешность усечения возникает из-за замены функции f(x) на интерполирующий ее многочлен. Погрешность округления возникает из-за того, что значение функции y_i в узлах интерполяции известно не точно, а приближенно, с некоторой погрешностью η.

Теорема 5.2:

$\varepsilon_{\text{усеч}}$ (с учётом знака) для формулы трапеции (5.12)

$$(5.15) \quad \varepsilon_{\text{усеч}} = -\frac{h^2}{12} f''(C) \quad C \in (x_0, x_2)$$

Комментарии: если $f'' > 0$, то $\varepsilon_{\text{усеч}} < 0$ ($\varepsilon_{\text{усеч}} = I_{\text{точное}} - I_{\text{приближенное}}$)



Теорема 5.3:

$\varepsilon_{\text{усеч}}$ (с учётом знака) для формулы Симпсона:

$$(5.16) \quad \varepsilon_{\text{усеч}} = -\frac{h^5}{90} f'''(C) \quad C \in (x_0, x_2)$$

Замечания:

Из (5.15) видно – формула трапеций выдаёт правильный результат ($\varepsilon_{\text{усеч}} = 0$), если f – многочлен первой степени (т.к. $f'(x)=0$). Этого следовало ожидать, т.к. при выведении формулы трапеции мы заменяли $f(x)$ на И.М. её первой степени, который совпадает с $f(x)$. По этой причине логично ожидать, что $\varepsilon_{\text{усеч}}$ для формулы Симпсона будет нулевой, если f – многочлен второй степени (т.к. в формуле Симпсона происходит интерполяция по трём точкам). Как мы видим из (5.16) формула Симпсона будет верна не только для многочлена третьей степени, т.к. $f'''(x)=0$.

6.4. Общие формулы трапеции и Симпсона численного интегрирования.

Если требуется найти $\int f(x)$ на большом промежутке, то мы разбиваем этот интервал на множество меньших интервалов, на каждом из которых применяем соответствующую формулу Ньютона – Котеса (трапеции или Симпсона).

Общая формула трапеций.

$$\begin{aligned} \int_{x_0}^{x_n} f(x) dx &= \int_{x_0}^{x_1} f(x) dx + \int_{x_1}^{x_2} f(x) dx + \dots + \int_{x_{n-1}}^{x_n} f(x) dx = \left\{ \begin{array}{l} \text{используем формулу} \\ \text{трапеции на каждом} \\ \text{интервале} \end{array} \right\} \approx \\ &\approx (x_1 - x_0) \left[\frac{y_0 + y_1}{2} \right] + (x_2 - x_1) \left[\frac{y_1 + y_2}{2} \right] + \dots + (x_n - x_{n-1}) \left[\frac{y_{n-1} + y_n}{2} \right] = \\ &= h \left[\frac{1}{2} (y_0 + y_n) + y_1 + y_2 + \dots + y_{n-1} \right] \approx \int_{x_0}^{x_n} f(x) dx \end{aligned} \quad (5.17)$$

Общая формула Симпсона.

Т.к. в формуле Симпсона участвует интеграл от x_0 до x_2 , то, при разбиении исходного участка, мы группируем интервалы попарно и поэтому $n=2k$ (общее количество участков должно быть чётное).

$$\int_{x_0}^{x_n} f(x)dx = \int_{x_0}^{x_2} f(x)dx + \int_{x_2}^{x_4} f(x)dx + \dots + \int_{x_{n-2}}^{x_n} f(x)dx = \begin{cases} n = 2k - \\ \text{четное} \\ \text{число} \end{cases} \approx$$

$$\approx 2h \left[\frac{1}{6} y_0 + \frac{2}{3} y_1 + \frac{1}{6} y_2 \right] + 2h \left[\frac{1}{6} y_2 + \frac{2}{3} y_3 + \frac{1}{6} y_4 \right] + \dots + 2h \left[\frac{1}{6} y_{n-2} + \frac{2}{3} y_{n-1} + \frac{1}{6} y_n \right] =$$

$$= \frac{h}{3} [(y_0 + y_n) + 4(y_1 + y_3 + y_5 + \dots + y_{2n-1}) + 2(y_2 + y_4 + \dots + y_{n-2})] \approx \int_{x_0}^{x_n} f(x)dx \quad (5.18)$$

6.5. Погрешности общих формул трапеции и Симпсона.

Погрешности усечения общей формулы трапеции и общей формулы Симпсона состоят из суммы погрешностей усечения формулы трапеции и формулы Симпсона на каждом интервале.

$\varepsilon_{\text{усеч}}$ для общей формулы трапеции (5.17):

$$\varepsilon_{\text{усеч}}([x_0, x_n]) = \varepsilon_{\text{усеч}}([x_0, x_1]) + \varepsilon_{\text{усеч}}([x_1, x_2]) + \dots + \varepsilon_{\text{усеч}}([x_{n-1}, x_n]) = -\frac{h^3}{12} f''(C_1) - \frac{h^3}{12} f''(C_2) +$$

$$+ \dots + \frac{h^3}{12} f''(C_n) = -\frac{h^3}{12} (f''(C_1) + \dots + f''(C_n)) = \left\{ \text{по теореме о среднем} \right\} = -\frac{h^2}{12} f''(C)n =$$

$$= -(b-a) \frac{h^2}{12} f''(C) = \varepsilon_{\text{усеч}} \quad (hn=b-a) \quad (5.19)$$

Аналогично выводим $\varepsilon_{\text{усеч}}$ для общей формулы Симпсона (5.18):

$$\varepsilon_{\text{усеч}}([x_0, x_n]) = \varepsilon_{\text{усеч}}([x_0, x_2]) + \varepsilon_{\text{усеч}}([x_2, x_4]) + \dots + \varepsilon_{\text{усеч}}([x_{n-2}, x_n]) = -\frac{h^5}{90} f'''(C_1) - \frac{h^5}{90} f'''(C_2) +$$

$$+ \dots + \frac{h^5}{90} f'''(C_n) = -\frac{h^5}{90} \sum_{i=1}^k f'''(C_i) = \left\{ \text{по теореме о среднем} \right\} = -\frac{h^5}{90} f'''(C)k =$$

$$= -(b-a) \frac{h^4}{90} f'''(C) = \varepsilon_{\text{усеч}} \quad (kh = \frac{b-a}{2}) \quad (5.20)$$

Т.к. местоположение точек C нам не известно, то (5.19) и (5.20) мы заменяем на оценки сверху (2.3 тах соответствующих производных).

Для формулы трапеции:

$$(5.21) \quad \varepsilon_{\text{усеч}} \leq \frac{h^2}{12} (b-a) M_2, \text{ где } M_2 = \max |f^{(3)}(t)|$$

$$(5.22) \quad \varepsilon_{\text{усеч}} \leq \frac{h^4}{180} (b-a) M_4, \text{ где } M_4 = \max |f^{(5)}(t)|$$

Погрешность округления общей формулы трапеции и общей формулы Симпсона:

Для формулы трапеции:

$$\int_{x_0}^{x_n} f(x)dx = h \left(\frac{1}{2} (y_0 + y_n) + y_1 + y_2 + \dots + y_{n+1} \right)$$

$$\left\{ (y_0 + y_n) \rightarrow 2\eta \quad ; \quad \frac{1}{2} (y_0 + y_n) \rightarrow \eta \right\}$$

$$\varepsilon_{\text{окр}} = n \cdot \eta \cdot h = (b-a) \cdot \eta$$

Аналогично для формулы Симпсона:

$$\frac{h}{3} [(y_0 + y_n) + 4(y_1 + y_3 + y_5 + \dots + y_{2n-1}) + 2(y_2 + y_4 + \dots + y_{n-2})]$$

$$\left\{ \begin{array}{l} (y_0 + y_n) \rightarrow 2\eta \quad ; \quad (y_2 + y_4 + \dots + y_{n-2}) \rightarrow (k-1)\eta \quad ; \quad 4(y_1 + y_3 + y_5 + \dots + y_{2n-1}) \rightarrow k\eta \end{array} \right\}$$

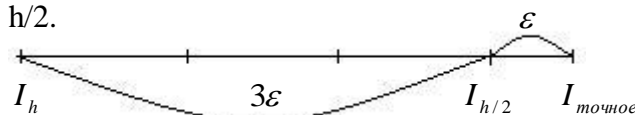
$$\varepsilon_{окр} = (b-a) \cdot \eta$$

6.6. Метод двойного пересчёта для оценки погрешности численного интегрирования.

При практических вычислениях часто бывает затруднительно оценить погрешность усечения формулы трапеции или формулы Симпсона из-за того, что надо находить $\max f''$ или $f^{(4)}$. В этом случае используется метод двойного пересчета.

Заметим, что при уменьшении шага в 2 раза, $\varepsilon_{усеч}$ в формуле трапеций уменьшается в 4 раза, а в формуле Симпсона в 16 раз.

Поэтому поступим следующим образом: вычислим интеграл на (a,b) дважды – с шагом h и h/2.



(рис.1)

если $I_h - I_{h/2} < 3\varepsilon$, то $|I_{h/2} - I_{\text{точное}}| < \varepsilon$. И поэтому в начале точное значение интеграла можно взять $I_{h/2}$ (оно будет найдено с заданной точностью).

Итак, при вычислении интеграла с помощью двойного пересчёта поступаем следующим образом: $I_h - I_{h/2} < 3\varepsilon$, $I_{\text{точное}} = I_{h/2}$

если точность не достигнута, то шаг h уменьшаем в 2 раза, находим $\int I_h \cdot I_{h/2}$ и так далее, пока точность не будет достигнута.

Замечание:

Формула трапеции имеет 2-ой порядок точности, т.к. в оценке $\varepsilon_{усеч}$ для глобальных формул трапеции (имеется в виду глобальный вариант формулы (т.е. применяем формулу на одном и том же интервале)) h^2 и поэтому при уменьшении шага в k раз - $\varepsilon_{усеч}$ уменьшается в k^2 раз.

Формула Симпсона имеет 4-ый порядок точности, т.к. h^4 .

6.7. Метод коррекции в двойном пересчёте.

Как видно из графика (рис.1), при двойном пересчёте в качестве $I_{\text{точного}}$ выгодно использовать не $I_{h/2}$, а:

$$I = I_{h/2} + 1/3(I_{h/2} - I_h) = I_{\text{кор}}$$

После коррекции по этой формуле точность метода возрастает на порядок, т.е. метод трапеции будет не второго, а третьего; а Симпсона – пятого порядка точности.

При использовании формулы Симпсона в методе двойного пересчета вместо 3ε будет 15ε .

Если $|I_h - I_{h/2}| < 15\varepsilon$, то $I_{h/2}$ – значение интеграла с погрешностью, не больше ε .

Тема 6: Численные методы решения дифференциальных уравнений и систем дифференциальных уравнений (ДУ и СДУ).

П.1. Постановка задачи.

Необходимо решить ДУ и СДУ на некотором наперёд заданном интервале с наперёд заданной точностью, либо оценить погрешность, которая найдена решением. В ЧМ мы ищем только частные решения ДУ.

П.2. Простейший вариант задачи. Простейший метод её решения – метод Эйлера.

Имеем ДУ 1-го порядка, а вместе с ним одно начальное условие:

$$\begin{cases} f(x, y, y') = 0 & - \text{ДУ} \\ y(x_0) = y_0 & - \text{начальное условие} \end{cases} \quad (\text{Задача Коши})$$

В дальнейшем всегда будем рассматривать ДУ, разрешенное относительно старшей производной, т.е. вида:

$$\begin{cases} y'(x) = f(x, y) \\ y(x_0) = y_0 \end{cases} \quad (6.1)$$

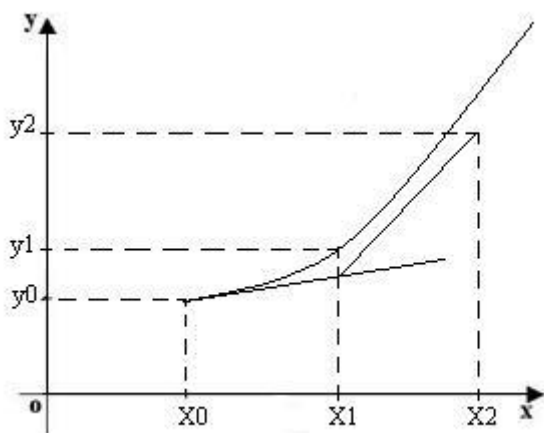
Общая идея всех методов численного решения ДУ и СДУ:

Фиксируем шаг h и будем находить по некоторым специальным формулам $y(x_0)$ - задан, $y(x_1), y(x_2), \dots, y(x_n)$, где x_i - равностоящие точки, а x_0, x_n - границы интервала $[a, b]$, на котором нам необходимо найти решение ДУ. При этом, необходимо брать шаг h достаточно малым, с тем, чтобы погрешность была невелика.

Простейший метод решения ДУ – метод Эйлера:

Заметим, что $y'(x_0) = f(x_0, y_0)$ - величина нам известная. Заменим неизвестное нам решение ДУ на касательную, а именно: $y(x_1) = y(x_0) + y'(x_0)h = y_0 + f(x_0, y_0)h$
В общем виде: $y_{i+1} = y_i + f(x_i, y_i) \cdot h$ (формула Эйлера).

Геометрическая интерпретация метода Эйлера:



Локальная погрешность метода Эйлера: $\text{const} \cdot h^2$

П.3. Простейшая модификация метода Эйлера – метод Рунге-Кутты 2-го порядка.

Заменяем приращение функции на первом шаге $y(x_1) - y(x_0)$ не на $y'(x_0) \cdot h$, как делали в методе Эйлера, а на более точное значение - на значение производной в середине интервала $h \cdot y'(x_0 + \frac{h}{2})$. А для того, чтобы найти $y'(x_0 + \frac{h}{2}) = f(x_0 + \frac{h}{2}, y(y_0 + \frac{h}{2}))$.

Заменяем $y(y_0 + \frac{h}{2}) \approx y(x_0) + \frac{h}{2} \cdot f(x_0, y_0)$

Окончательно получаем следующую формулу:

$$\begin{cases} y_{i+1} = y_i + f(x_i + \frac{h}{2}, y_{i+1/2}^*) \\ \text{где } y_{i+1/2}^* = y_i + \frac{h}{2} \cdot f(x_i, y_i) \end{cases} \quad (6.3) \quad \text{Формула Рунге-Кутты 2го порядка с усреднением по времени.}$$

Метод Рунге-Кутты 2го порядка с усреднением по производной.

Заменяем приращение:

$$y(x_1) - y(x_0) = \int_{x_0}^{x_1} y'(x) dx \approx \frac{h}{2} (y'(x_1) + y'(x_0))$$

$$y'(x_1) = f(x_1, y_1) \approx y_0 + h \cdot f(x_0, y_0)$$

Окончательно получим следующую формулу:

$$\begin{cases} y_{i+1} = y_i + \frac{h}{2} (f(x_i, y_i) + f(x_i + h, y_{i+1}^*)) \\ \text{где } y_{i+1}^* = y_i + \frac{h}{2} f(x_i, y_i) \end{cases} \quad (6.4) \quad \text{Метод Рунге-Кутты 2го порядка с усреднением по производной.}$$

Локальная погрешность методов (6.3) и (6.4) $\text{const} \cdot h^3$

II.4. Сведение дифференциальных уравнений высших порядков к системе дифференциальных уравнений и её решение.

Рассмотрим дифференциальное уравнение $n^{\text{ого}}$ порядка, разрешенное относительно старшей производной и Задачу Коши для данного уравнения:

$$\begin{cases} y^{(n)}(x) = f(x, y, y', y'', \dots, y^{(n-1)}) & - \text{ДУ} \\ y(x_0) = y_0 \\ y'(x_0) = y_0' \\ \dots\dots\dots \\ y^{(n-1)}(x_0) = y_0^{n-1} \end{cases} \quad (6.5) \quad \text{- начальные условия.}$$

Чтобы свести З.К.(6.5) для ДУ $n^{\text{ого}}$ порядка к СДУ $1^{\text{ого}}$ порядка, поступаем следующим образом:

введём вектор-функцию $Y(x) = \begin{pmatrix} y(x) \\ y'(x) \\ \dots \\ y^{(n-1)}(x) \end{pmatrix}$, тогда З.К. (6.5) для ДУ $n^{\text{ого}}$ порядка сводится к

$$\begin{cases} Y'(x) = F(X, Y(x)) \\ Y(x_0) = Y_0 \end{cases} \quad (6.6) \quad \text{где } F(X, Y(x)) = \begin{pmatrix} y'(x) \\ y''(x) \\ \dots \\ f(x, y, y', \dots, y^{(n-1)}) \end{pmatrix}; \quad Y(x_0) = \begin{pmatrix} y_0 \\ y_0' \\ \dots \\ y_0^{(n-1)} \end{pmatrix}$$

Итак, вместо З.К. (6.5) для ДУ $n^{\text{ого}}$ порядка мы получили З.К. (6.6) для СДУ $1^{\text{ого}}$ порядка, а её мы можем решить любым известным нам методом (Эйлера, Рунге-Кутта,...) с заменой в этих формулах скалярных величин y, f на векторные Y, F .

Пример сведения ДУ $n^{\text{ого}}$ порядка к СДУ $1^{\text{ого}}$ порядка и нахождение решения по методу Эйлера:

$$\begin{cases} y''x - y' = y - x^2 \\ y(1) = 2 \\ y'(1) = -4 \end{cases}$$

Имеем ДУ 2го порядка, сводим к СДУ 1го порядка для 2х уравнений:

$$\text{Вводим } Y = \begin{pmatrix} y \\ y' \end{pmatrix}$$

$$Y' = F(x, Y(x)) = \begin{pmatrix} y' \\ \frac{y + y' + x^2}{x} \end{pmatrix}$$

фиксируем шаг: $h=0,1$

$$Y_0 = \begin{pmatrix} 2 \\ -4 \end{pmatrix}$$

$$Y_1 = Y_0 + hF(x_0, Y_0) = \begin{pmatrix} 2 \\ -4 \end{pmatrix} + 0,1 \cdot \begin{pmatrix} -4 \\ \frac{2 - 4 + 1^2}{1} \end{pmatrix} = \begin{pmatrix} 2 \\ -4 \end{pmatrix} + \begin{pmatrix} -0,4 \\ -0,1 \end{pmatrix} = \begin{pmatrix} 1,6 \\ -4,1 \end{pmatrix}$$

Аналогичным образом находим Y_2

$$Y_2 = Y_1 + hF(x_1, Y_1) = \begin{pmatrix} 1,6 \\ -4,1 \end{pmatrix} + 0,1 \cdot \begin{pmatrix} -4,1 \\ \frac{1,6 - 4,1 + (1,1)^2}{1,1} \end{pmatrix} = \begin{pmatrix} 1,6 \\ -4,1 \end{pmatrix} + \begin{pmatrix} -0,41 \\ -0,1722 \end{pmatrix} = \begin{pmatrix} 1,19 \\ -4,2722 \end{pmatrix}$$

Метод Рунге-Кутта 4го порядка.

Наиболее применяемым методом решения ДУ и СДУ является метод Рунге-Кутта 4го порядка.

Формулы метода Рунге-Кутта 4го порядка:

$$k_1 = f(x_i, y_i)$$

$$k_2 = f\left(x_i + \frac{h}{2}, y_i + \frac{h}{2}k_1\right) \quad (6.7)$$

$$k_3 = f\left(x_i + \frac{h}{2}, y_i + \frac{h}{2}k_2\right)$$

$$k_4 = f(x_i + h, y_i + hk_3)$$

$$y_{i+1} = y_i + \frac{h}{6}(k_1 + 2k_2 + 2k_3 + k_4)$$

в векторной форме данной формулы, величины y, f, k заменяют на Y, F, K .

П.5. Локальные и глобальные погрешности одношаговых методов решения ДУ (метода Эйлера и методов Рунге-Кутта 2го, 4го порядка).

Теорема 6.1:

Если локальная погрешность метода $counst \cdot h^{p+1}$, то глобальная $counst \cdot h^p$.

Комментарии:

как и при численном интегрировании, при переходе от локальной погрешности к глобальной, точность метода уменьшается на порядок. (6.8):

Методы	Локальная	Глобальная
Эйлер	$\text{const} \cdot h^2$	$\text{const} \cdot h$
Р.-К. 2го порядка по времени	$\text{const} \cdot h^3$	$\text{const} \cdot h^2$
Р.-К. 2го порядка по производной	$\text{const} \cdot h^3$	$\text{const} \cdot h^2$
Р.-К. 4го порядка	$\text{const} \cdot h^5$	$\text{const} \cdot h^4$

Как и при численном интегрировании, порядок метода – степень h в глобальной погрешности.

II.6. Многошаговые методы решения ДУ и СДУ.

Все рассмотренные ранее методы – одношаговые, т.к. для нахождения y_{i+1} мы использовали только лишь значения y_i с предыдущего шага. В многошаговых методах для нахождения y_{i+1} используется не только лишь одно y_i , но и предыдущие значения. В k -шаговом методе используются значения с k предыдущих шагов. Многошаговые методы, как правило, дают лучший результат, чем одношаговые, в силу того, что более устойчивы к вычислительным погрешностям. Многошаговых методов много, самый распространенный среди них – метод Милна.

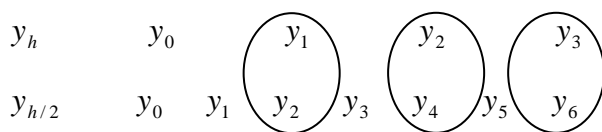
Формулы метода Милна:

$$\begin{cases} y_{i+1}^* = y_{i-3} + \frac{4h}{3}(2f(x_{i-2}, y_{i-2}) - f(x_{i-1}, y_{i-1}) + 2f(x_i, y_i)) \\ y_{i+1} = y_{i-1} + \frac{h}{3}(f(x_{i-1}, y_{i-1}) + 4f(x_i, y_i) + f(x_{i+1}, y_{i+1}^*)) \end{cases} \quad (6.9)$$

Метод Милна – 4х шаговый (т.к. использует 4 предыдущих значения) и имеет 4-ый порядок точности. Перед применением метода Милна нам надо знать 4у, следовательно, необходимо сделать хотя бы 3 шага каким-нибудь одношаговым методом.

II.7. Оценка погрешности решения ДУ и СДУ методом двойного пересчета. Коррекция решения.

Используя такую же идею, как и в численном интегрировании, находим решение ДУ на $[a, b]$ дважды с шагом h и с шагом $h/2$. Получим следующую картину:



Сравниваем попарно, если расхождение между $|y_{h(k)} - y_{h/2(k)}| < 3\varepsilon$ для метода 2го порядка, $|y_{h(k)} - y_{h/2(k)}| < 15\varepsilon$ для метода 4го порядка, то в качестве точного решения берём $y_{h/2}$. Если же точность не достигнута, то шаг h уменьшаем вдвое и т.д., пока она не будет достигнута.

Метод двойного пересчёта при решении ДУ и СДУ практически единственный имеет возможность для оценки погрешностей, так как иные формулы очень сложны и требуют оценок различных производных.

Как и при ЧИ, при решении ДУ и СДУ после 2го пересчёта в качестве точного решения выгодно брать не $Y_{h/2}$, а $Y_{кор}$.

$$Y_{кор} = Y_{h/2} + \frac{1}{3}(Y_{h/2} - Y_h) \text{ - для второго порядка}$$

Метод двойного пересчёта применим не только лишь при ЧИ, при решении ДУ и СДУ, но и при решении других численных методов.

П.7. Краевые задачи для дифференциальных уравнений.

Выше рассматривалось решение ДУ и СДУ с начальными условиями, заданными в одной точке, так называемую задачу Коши, но для ДУ высших порядков часто бывает необходимо решить не з. Коши, а так называемую краевую задачу, т.е. начальные условия, которые заданы в разных точках.

Рассмотрим простейшую краевую задачу для ДУ 2го порядка:

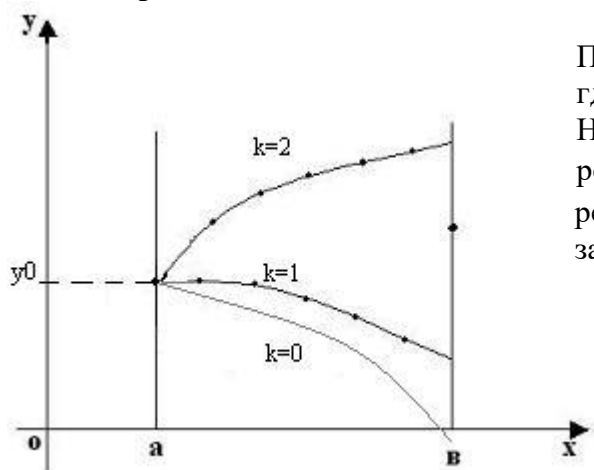
$$\begin{cases} y'' = f(x, y, y') \\ y(a) = y_0 \\ y(b) = y_1 \end{cases} \quad (6.10)$$

А мы умеем решать:

$$\begin{cases} y'' = f(x, y, y') \\ y(a) = y_0 \\ y'(a) = y'_0 = k \end{cases} \quad (6.11),$$

В (6.11) нам известно $y'(a)$, поэтому для решения задачи (6.10) мы будем подбирать $y'(a)$ в (6.11), с тем, чтобы $y(b) = y_1$

Метод стрельб



После пристрелки и определения интервала $[a, b]$, где идёт смена знака, запускаем МПД или МХ. На практике это выглядит так, как будто мы решаем уравнение $q(k) = y_1$, где $q(k)$ возвращает решение задачи Коши (6.11) в точке b при заданном k .

П.9. Что делать, если ДУ не может быть разрешено относительно старшей производной?

Так как ДУ не может быть решено относительно старшей производной, то тогда на каждом шаге решаем нелинейное уравнение относительно $y^{(n)}$ (все остальные неизвестные $y, y', y'', \dots, y^{(n-1)}$ к этому моменту уже известны).

Решать уравнение относительно старшей производной любым методом (Хорд, МПД, Ньютона).

Замечание:

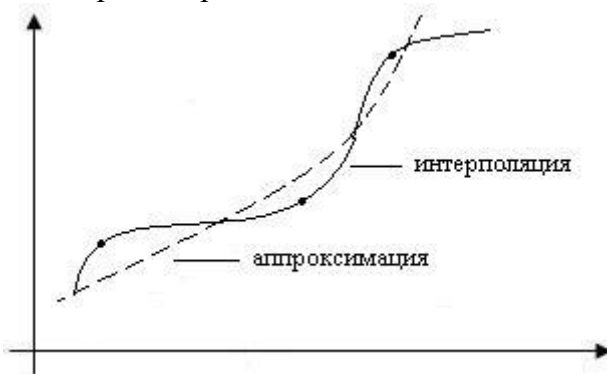
Таким образом, если ДУ не разрешается относительно старшей производной, то у нас возникает дополнительный цикл (самый внутренний) при написании программы.

Тема 7: Аппроксимация.

П.1. Постановка задачи аппроксимации.

Пусть некоторая функция $y=f(x)$ задана в точках x_0, x_1, \dots, x_n . Фиксируем некоторый класс функций G .

Задача аппроксимации: выбрать среди функций $g \in G$ некоторую функцию g' , которая лучше всего приближает функцию f в узлах x_i . В отличие от интерполяции при аппроксимации не требуется, чтобы $g(x_i)=y_i$, а требуется лишь $g(x_i) \approx y_i$. Обычно аппроксимацию применяют, когда значения функции f были известны не точно, а с некоторой погрешностью.



Конкретизация задачи аппроксимации.

Для оценки близости функции g и f составляется вектор невязок (погрешности)

$$R=(g(x_0)-f(x_0), g(x_1)-f(x_1), \dots, g(x_n)-f(x_n)).$$

Очевидно, что мы будем идти к тому, чтобы норма R была намного меньше, при этом можно работать либо:

1. $\| \cdot \|_{\infty}$ - минимизируется максимальное отклонение.
2. $\| \cdot \|_1$ - минимизируем сумму отклонений.
3. $\| \cdot \|_2$ - минимизируем сумму квадратов отклонений.

Легко реализуются вычисления именно для 2ой нормы.

В качестве класса аппроксимирующей функции G , рассмотрим всевозможные линейные комбинации базисных функций g_0, g_1, \dots, g_n .

$$G(x) = \left\{ \sum_{i=0}^k a_i y_i(x) \quad (a_i \in R) \right\} \text{ базисные функции фиксированы.}$$

П.2. Метод наименьших квадратов.

Фиксирован набор функций: $g_0(x), g_1(x), \dots, g_k(x)$ даны $(x_0, y_0), (x_1, y_1), \dots, (x_n, y_n)$.

$$g(x) = \sum_{j=0}^k a_j g_j(x) \text{ - линейная комбинация функции } g_j, \quad y_i = f(x_i) \quad 0, \dots, n;$$

фиксируем набор:

$$R=(y_0-g(x_0), \dots, y_n-g(x_n))$$

$$S(a_0, \dots, a_k) = \sum_{i=0}^n (y_i - g(x_i))^2 = \sum_{i=0}^n \left(y_i - \sum_{j=0}^k a_j g_j(x_i) \right)^2 \rightarrow \min.$$

Для нахождения минимума функции S от $(k+1)$ переменных, нам необходимо все её частные производные приравнять к 0. При этом получаем систему: $(k+1)$ уравнения для $(k+1)$ переменных (a_0, \dots, a_k) .

$$0 = \frac{\partial S}{\partial a_p} = \sum_{i=0}^n 2 \left(y_i - \sum_{j=0}^k a_j g_j(x_i) \right) g_p(x_i)$$

$$\sum_{i=0}^n \sum_{j=0}^k a_j g_j(x_i) g_p(x_i) = \sum_{i=0}^n y_i g_p(x_i)$$

$$\sum_{j=0}^k a_j \sum_{i=0}^n g_j(x_i) g_p(x_i) = \sum_{i=0}^n y_i g_p(x_i) \quad , p = \overline{0, k} \quad (7.1)$$

Получили СЛАУ (7.1). Эту систему можно решить методом Гаусса – она всегда имеет единственное решение в случае, если набор базисных функций был линейно независим в узлах x_i , то матрица СЛАУ (7.1) невырожденная, поэтому решение будет существовать и единственное.

Пример аппроксимации полиномами:

$$y(x) = a_0 + a_1 x + a_2 x^2$$

базисные функции:

$$g_0(x) = 1$$

$$g_1(x) = x$$

$$g_2(x) = x^2$$

для аппроксимации функциями такого вида, нам необходимо решить СЛАУ 3 на 3:

$$\begin{cases} C_{00}a_0 + C_{01}a_1 + C_{02}a_2 = b_0 \\ C_{10}a_0 + C_{11}a_1 + C_{12}a_2 = b_1 \\ C_{20}a_0 + C_{21}a_1 + C_{22}a_2 = b_2 \end{cases}$$

$$C_{00} = \sum_{i=1}^n g_0(x_i) g_0(x_i) = \sum_{i=1}^n 1 \cdot 1$$

$$C_{01} = \sum_{i=1}^n g_0(x_i) g_1(x_i) = \sum_{i=1}^n x_i$$

$$C_{10} = \sum_{i=1}^n g_1(x_i) g_0(x_i) = \sum_{i=1}^n x_i$$

$$C_{11} = \sum_{i=1}^n g_1(x_i) g_1(x_i) = \sum_{i=1}^n x_i^2$$

$$C_{02} = \sum_{i=1}^n g_0(x_i) g_2(x_i) = \sum_{i=1}^n x_i^2$$

$$C_{12} = \sum_{i=1}^n g_1(x_i) g_2(x_i) = \sum_{i=1}^n x_i^3$$

$$C_{20} = \sum_{i=1}^n g_2(x_i) g_0(x_i) = \sum_{i=1}^n x_i^2$$

$$C_{21} = \sum_{i=1}^n g_2(x_i) g_1(x_i) = \sum_{i=1}^n x_i^3$$

$$C_{22} = \sum_{i=1}^n g_2(x_i) g_2(x_i) = \sum_{i=1}^n x_i^4$$

Тема 8: Нелинейная оптимизация.

Метод градиента (метод наискорейшего спуска).

П.1. Сведение системы линейных уравнений к задаче нелинейной оптимизации (ЗНО) и наоборот.

Постановка задачи ЗНО:

Найти $f(X) = f(x_1, \dots, x_n)$ (8.1) минимум или максимум в некоторой области D.

Как мы помним из мат. анализа, следует приравнять частные производные к нулю.

Таким образом, ЗНО (8.1) свели к СЧУ (8.2)

$$\frac{\partial f}{\partial x_i} = 0 \quad i = \overline{0,1} \quad (8.2) \quad \text{n нелинейных уравнений.}$$

Обратное: пусть дана СЧУ

$$\begin{cases} y_1(x_1, \dots, x_n) = 0 \\ \dots\dots\dots \\ y_n(x_1, \dots, x_n) = 0 \end{cases} \iff S(x_1, \dots, x_n) = \sum_{i=1}^n g_i^2(x_1, \dots, x_n) \rightarrow \min$$

(S достигает своего минимума в точке, где все i-ые зануляются)

П.2. Метод градиента.

Наша задача найти минимум функций n переменных без ограничений на X.

Общая идея метода:

Возьмём некоторую стартовую точку X (желательно, чтобы она была достаточно близка к минимуму функции).

Пусть $g(x_1, \dots, x_n)$ - гладкая функция, тогда вектор градиента ∇f в точке x_0 :

$$\text{grad}f(X^{(0)}) = \left(\frac{\partial f}{\partial x_1}, \dots, \frac{\partial f}{\partial x_n} \right) \Big|_{X^{(0)}} \quad \text{показывает направление наискорейшего роста функции.}$$

Соответственно вектор $-\text{grad}f(X^{(0)})$ показывает направление наискорейшего спуска.

Пойдём вдоль этого вектора, рассмотрим функцию

$$h_0(t) = f(X^{(0)} - t \text{grad}f(X^{(0)})), \quad t \geq 0$$

уравнение луча, исходящего из $X^{(0)}$ в направлении $-\text{grad}$

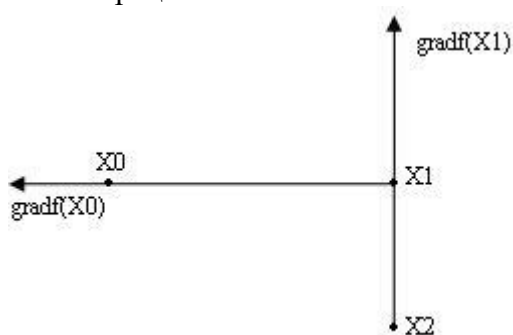
Идём вдоль этого луча до тех пор, пока $h_0(t)$ не начнёт возрастать, т.е. не достигнет своего минимума. В этот момент (в точке $X^{(1)}$) мы остановимся и повторим такую же процедуру (т.е. пойдём из точки $X^{(1)}$ в направлении $-\text{grad}$, т.е. рассмотрим

$$h_1(t) = f(X^{(1)} - t \text{grad}f(X^{(1)})) \text{ и т.д.})$$

Данную процедуру повторяем до тех пор, пока не достигнем заданной точности

(применяем универсальный критерий прерывания $|X^{(n+1)} - X^{(n)}| < \varepsilon$)

Иллюстрация:

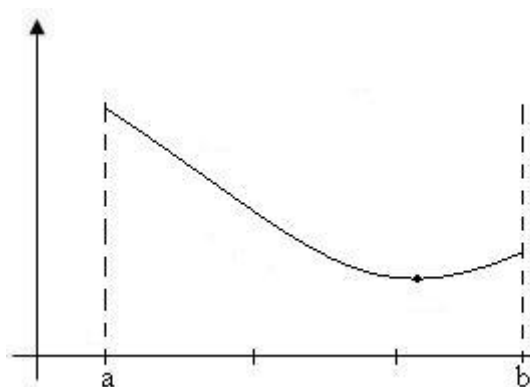


Итак, с помощью метода градиента нам удастся многомерную ЗНО свести к одномерной ЗНО, т.е. нам необходимо научиться находить минимум функций одной переменной (на первом шаге минимум $h_0(t)$, на втором шаге $h_1(t)$ и т.д.)

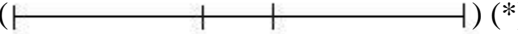
П.3. Решение одномерной ЗНО.

Нам необходимо найти минимум некоторой функции f (одной переменной) на интервале $[a,b]$. Будем считать, что f на данном интервале имеет ровно один минимум.

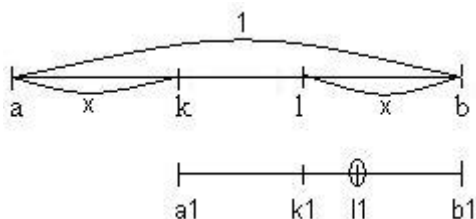
Простейший метод нахождения точки минимума – метод сечений:



бессмысленно).

() (*)

На самом же деле выгоднее делить интервал не таким образом, а по методу золотого сечения.



Пропорции деления (они сохраняются) были подобраны таким образом, что при переходе к новому интервалу нам необходимо вычислять значения f не в 2х новых точках, а в 1-ой, хотя при этом интервал поиска сокращается не в два раза (как было на предложенной схеме (*)), а чуть меньше, следовательно, этот вариант оказывается выгоднее, т.к. для достижения заданной точности нам придётся меньше раз вычислять значение f . 4 точки на первом шаге и по 1-ой новой точке на каждом последующем шаге, а не в 2х.

Найдём пропорции золотого сечения:

$$\frac{b-l}{b-k} = \frac{b-k}{b-a}$$

$$\frac{x}{1-x} = \frac{1-x}{1}$$

$$x^2 - 2x + 1 = 0$$

$$x_{1,2} = \frac{3 \pm \sqrt{5}}{2}; \quad x_2 = 0,382$$

Итак, в алгебре метода золотого сечения заложены пропорции: **0,382 ; 0,236 ; 0,382**.

Тема 9: Метод Монте-Карло.

П.1.Особенности метода Монте-Карло.

Главная особенностью метода Монте-Карло – то, что он в отличие от всех остальных методов вероятностный. Т.е., если во всех остальных методах мы знали, с вероятностью равной 1, что достигли заданной точности, то в методе Монте-Карло мы можем только лишь утверждать, что P – близка к единице, т.е. вероятность ошибиться, мала, но есть. Причём, всякий раз при новом запуске метода, результат будет другим, т.к. он носит вероятностный характер. Наша основная цель научиться оценить вероятность того, что найденное значение X , отличается от точного не более чем на ε ($P(|X - X_{\text{точн.}}| < \varepsilon)$).

Методом Монте-Карло можно решать многие задачи вычислительной математики: СЛАУ, ДУ, ЧИ и т.д.

Мы рассмотрим простейший пример применения метода Монте-Карло.

П.2.Метод Монте-Карло в ЧИ.

Обычные (невероятностные) методы ЧИ хорошо работают только лишь при интегрировании функций небольшого количества переменных. При росте количества переменных, число узлов интегрирования стремительно растёт (экспоненциально). Например, пусть необходимо найти интеграл 10-ти переменных с шагом $h=0,1$ по каждому измерению:

$$\int_0^1 \dots \int_0^1 f(x_1, \dots, x_n) dx_1 \dots dx_n \text{ итого } 11^{10} \text{ измерений – в подобном случае применяют метод}$$

Монте-Карло.

Рассмотрим первый вариант метода Монте-Карло:

$$I = \int_0^1 \dots \int_0^1 f(x_1, \dots, x_n) dx_1 \dots dx_n = \{ \text{среднему значению на кубе } [0,1]^n, \text{ как известно из мат.}$$

анализа среднее значение функции равно $\frac{\int_a^b f(x) dx}{(b-a)}$, следовательно, для функций многих

$$\text{переменных } \frac{\int_0^1 \dots \int_0^1 f(x_1, \dots, x_n) dx_1 \dots dx_n}{V([0,1]^n)} \}.$$

В то же время, как известно из теории вероятности, среднее эмпирическое значение при увеличении количества испытаний стремится к точному значению.

В нашем случае эмпирическое среднее (для N испытаний)

$$f_{cp} = \frac{1}{N} \sum_{i=1}^N f(X^{(i)}) = \frac{1}{N} \sum_{i=1}^N f(X_1^{(i)}, \dots, X_n^{(i)}) = I_{прибл.}, \text{ таким образом, для нахождения } I_{прибл.} \text{ мы}$$

N раз находим значение функции f в N точках $X^{(i)}$, каждая из которых имеет n координат, при этом, каждая координата – случайное число на отрезке [0,1]. Итак, random нам потребуется вызвать $n \cdot N$ раз.

При каждом запуске метода Монте-Карло мы будем получать новые значения $I_{прибл.}$, но все они $\approx I_{точ.}$

П.2. Второй вариант метода Монте-Карло (интегрирование не по n-мерному кубу, а по некоторой произвольной n-мерной области D).

Необходимо найти:

$$\int_D f(x_1, \dots, x_n) dx_1 \dots dx_n = \int_{\Pi} g(x_1, \dots, x_n) dx_1 \dots dx_n =$$

$$\left[\begin{array}{l} \text{где } \Pi - \text{прямоугольный параллелепипед ограничивающий область } D. \\ \Pi = [a_1, b_1] \cdot [a_2, b_2] \cdot \dots \cdot [a_n, b_n] \quad g(x_1, \dots, x_n) = \begin{cases} f(x_1, \dots, x_n) & \text{если } (x_1, \dots, x_n) \in D \\ 0 & \text{если } (x_1, \dots, x_n) \notin D \end{cases} \end{array} \right]$$

$$= \int_{a_1}^{b_1} \dots \int_{a_n}^{b_n} g(x_1, \dots, x_n) dx_1 \dots dx_n = \frac{1}{N} \sum_{i=1}^N g(x_1^{(i)}, \dots, x_n^{(i)})$$

Чтобы получить равномерное распределение на $[a_i, b_i]$, берем $\text{random}^*(b_i - a_i) + a_i$ - все остальные вычисления аналогичны случаю n-мерного куба.

Недостатки:

Основная проблема, что в предложенном выше методе мы не можем достоверно оценить вероятность отклонения $I_{\text{прибл.}}$ от $I_{\text{точн.}}$, мы знаем только лишь, что $M(I_{\text{прибл.}}) = I_{\text{точн.}}$, т.е. при бесконечном количестве испытаний $I_{\text{прибл.}} = I_{\text{точн.}}$. А оценить разброс $I_{\text{прибл.}}$ от $I_{\text{точн.}}$ мы не можем, т.к. не знаем дисперсию.

Оценить вероятность отклонения случайной величины от её мат. ожидания, можно с

помощью неравенства Чебышева $P(|I - I_{\text{прибл.}}| > \varepsilon) < \frac{D(I_{\text{прибл.}})}{\varepsilon^2}$ (9.1)

С ростом числа испытаний N , $D(I_{\text{прибл.}}) \xrightarrow{N \rightarrow \infty} 0$, а именно, если мы проведём N испытаний,

$$\text{то } D(I_{\text{прибл.}}) = \frac{1}{N} \sum_{i=1}^N f(X^{(i)}) = \frac{D(\eta)}{N} \quad (9.2)$$

Если в (9.1) подставить формулу (9.2), то получим $P(|I - I_{\text{прибл.}}| > \varepsilon) < \frac{D(\eta)}{N\varepsilon^2}$

η - не зависит от N и ε , зависит от g и области интегрирования D .

η можно оценить, используя различные способы, например, найти эмпирическое значение дисперсии.