ЧИСЛЕННЫЕ МЕТОДЫ МАТЕМАТИЧЕСКОГО МОДЕЛИРОВАНИЯ.
(ЛЕКЦИИ темы:4-8)
Тема 4: ПРИМЕНЕНИЕ ИНТЕРПОЛЯЦИИ
§1. Обратное интерполирование

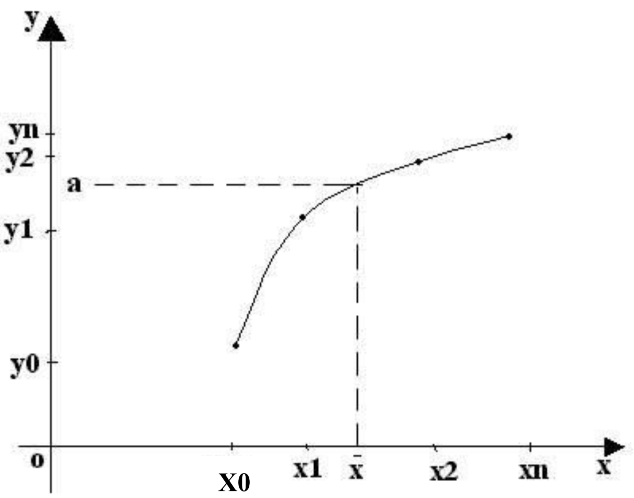
С помощью обратной интерполяции можно решать нелинейные уравнения вида: f(x)=a (f(x)=0).

Идея обратной интерполяции:

Пусть в точках $x_0, x_1, \dots, x_п$ заданы значения

$y_0=f(x_0)$, $y_1=f(x_1)\dots$, $y_п=f(x_п)$

Предположим, что функция f на участке $x_0, x_1, \dots, x_п$ монотонна – для определенности - монотонно возрастает, тогда существует обратная функция f^{-1} . Определим функцию $f^{-1}(y)$ и найдем значение функции $f(x_i)=y_i$ в точке a – это и будет решением нелинейного уравнения $f(x)=a$. Таким образом получена обычная задача интерполяции. Для решения этой задачи используется какой-либо метод интерполяции, например, интерполяция многочленами: метод Лагранжа.



Пример:

x	y
10	3
11	7
12	11
13	17

y(x)=10 (это ~ 16,5)

проводим интерполяционный многочлен через точки y_0, y_1, y_2, y_3 принимающие в этих точках значения указанные в таблице.

$$P_3(y)=\frac{(y-7)(y-11)(y-17)}{(3-7)(3-11)(3-17)}10+\frac{(y-3)(y-11)(y-17)}{(7-3)(7-11)(7-17)}15+$$

$$+\frac{(y-3)(y-7)(y-17)}{(11-3)(11-7)(11-17)}17+\frac{(y-3)(y-7)(y-11)}{(17-3)(17-7)(17-11)}20$$

чтобы найти корень, необходимо найти $P_3(10)=16,641$ – это решение уравнения.

§2.Численное дифференцирование функций
П.0 Постановка задачи численного дифференцирования

В узлах $x_0, x_1, \ldots, x_n$ заданы значения функции $y_0=f(x_0)\ldots y_n=f(x_n)$
Задача:
Найти производную f' и производные высших порядков от функции f в любой заданной точке x.

П.1 Формула численного дифференцирования (ЧД)

Общая идея ЧД: Неизвестную функцию f заменить на интерполирующую функцию g (то есть на функцию g: $g(x_i)=y_i$ и заменить $f'(x)$ на $g'(x)$) . В качестве интерполирующей функции рассмотрим только интерполяционный многочлен. В дальнейшем будем считать, что узлы $x_i=x_0+i\Delta x$, $i=\overline{0, n}$ равностоящие с шагом Δx. Рассмотрим первый интерполяционный многочлен Ньютона:

$$y(x) = y_0 + \frac{q}{1!}\Delta y_0 + \frac{q(q-1)}{2!}\Delta^2 y_0 + \frac{q(q-1)(q-2)}{3!}\Delta^3 y_0 + \frac{q(q-1)(q-2)(q-3)}{4!}\Delta^4 y_0 + \ldots$$

где: $q = \dfrac{x-x_0}{\Delta x}$, $\dfrac{dy}{dx} = \dfrac{dy}{dq} * \dfrac{dq}{dx} = \dfrac{dy}{dq} * \dfrac{1}{\Delta x}$

$$y'(x) = \frac{1}{\Delta x}\left(y_0 + q\Delta y_0 + \frac{q^2-q}{2}\Delta^2 y_0 + \frac{q^3-3q^2+2q}{6}\Delta^3 y_0 + \frac{q^4-6q^3+11q^2-6q}{24}\Delta^4 y_0 + \ldots\right)'_q =$$

$$= \boxed{\frac{1}{\Delta x}\left(0 + \Delta y_0 + \frac{2q-1}{2}\Delta^2 y_0 + \frac{3q^2-6q+2}{6}\Delta^3 y_0 + \frac{2q^3-9q^2+11q-3}{12}\Delta^4 y_0 + \ldots\right) = y'(x) \qquad (4.2)}$$

Как и в первой интерполяционной формуле Ньютона суммирование в формуле (4.2) можно прервать. При этом, если в формуле взято k слагаемых (не считая 0), то будет получена производная от интерполяционного многочлена, интерполирующая функцию не во всех точках, а только в крайних левых (k+1) точках. Продифференцировав формулу (4.2) по x еще один раз, получаем:

(4.3) $$y''(x) = \frac{1}{(\Delta x)^2}\left(\Delta^2 y_0 + (q-1)\Delta^3 y_0 + \frac{6q^2-18q+11}{12}\Delta^4 y_0 + \ldots\right) \qquad [5.2]$$

Если в (4.3) взять первые k слагаемых, то получим вторую производную от интерполяционного многочлена, проходящего через (k+2) точки $x_0, x_1, \ldots, x_{k+1}$.
Часто необходимо найти y' и y''.

(4.4) $$y'(x_0) = \frac{1}{\Delta x}\left(\Delta y_0 - \frac{1}{2}\Delta^2 y_0 + \frac{1}{3}\Delta^3 y_0 - \frac{1}{4}\Delta^4 y_0 + \ldots\right) \qquad [5.3]$$ доказательство 4 балла

(4.5) $$y''(x_0) = \frac{1}{(\Delta x)^2}\left(\Delta^2 y_0 - \Delta^3 y_0 + \frac{11}{12}\Delta^4 y_0 - \ldots\right) \qquad [5.4]$$

На практике удобнее дифференцировать не первый интерполяционный многочлен Ньютона (он односторонний), а центральную формулу Стирлинга, так как узлы интерполяции располагаются симметрично относительно начальной точки x_0 (см т.3 пар.1 п.4). Выпишем первые два слагаемых:

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

$$y'(x_0) = \frac{1}{\Delta x}(\frac{\Delta y_{-1} + \Delta y_0}{2}) = \frac{1}{\Delta x}(\frac{(y_0 - y_{-1}) + (y_1 - y_0)}{2}) =$$

$$\boxed{(4.6) \qquad = \frac{1}{2\Delta x}(y_1 - y_{-1}) = y'(x)} \qquad [5.5]$$

Формула (4.6) точнее, чем (4.2) с одним слагаемым. Аналогично на основе интерполяционного многочлена Стирлинга могут быть получены формулы:

$$(4.7) \qquad y'(x) = \frac{1}{12\Delta x}(y_{-2} - 8y_{-1} + 8y_1 - y_2) \qquad [5.6]$$

эта формула будет получена, если взять в интерполяционном многочлен Стирлинга первые три слагаемых и $x=x_0$ (т.е. $q=0$) :

$$(4.8) \qquad y''(x) = \frac{1}{(\Delta x)^2}(y_{-1} - 2y_0 + y_1) \qquad [5.7]$$

эта формула будет получена при двукратном дифференцировании первых трех слагаемых интерполяционного многочлен Стирлинга при $x=x_0$.

П.2 Оценка погрешностей численного дифференцирования

При численном дифференцировании, как и при интерполяции, возникает два типа погрешностей. 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 y_i известны не точно, а с некоторой погрешностью η. Оценим погрешность усечения.

Теорема 4.1

Погрешность усечения при численном дифференцировании (при использовании производной от первой интерполяционной формулы Ньютона – формула (4.2)), при суммировании k-слагаемых имеет следующую оценку:

$$(r_\kappa, (x))\,|^1_{x=x_0} = (f(x) - P_\kappa(x))\,|^1_{x=x_0} = (-1)^\kappa \frac{(\Delta x)^\kappa}{\kappa + 1} f^{(\kappa+1)}(\xi) \qquad (4.9)$$

(погрешность усечения можно оценить только в узлах интерполяции)

где $\xi \in [x_0, x_\kappa]$.

Доказательство:

Вспомним оценку остаточного члена первой интерполяционной формулы Ньютона

$$r_\kappa(x) = \frac{f^{(\kappa+1)}(\xi)}{(\kappa+1)!}(x - x_0)(x - x_1)...(x - x_\kappa)$$

Дифференцируем по x:

$$(r_\kappa(x))' = \frac{1}{(\kappa+1)!}\left[(f^{(\kappa+2)}(\xi))\,|_x^1 *(x-x_0)(x-x_1)...(x-x_\kappa) + f^{(\kappa+1)}(\xi)((x-x_0)(x-x_1)...(x-x_\kappa))\,|_x^1\right] =$$

$$= \frac{1}{(\kappa+1)!}[(f^{(\kappa+2)}(\xi))\frac{d\xi}{dx}(x-x_0)(x-x_1)...(x-x_\kappa) + f^{(\kappa+1)}(\xi)\{(x-x_0)_x^1(x-x_1)(x-x_2)...(x-x_\kappa) +$$

$$+(x-x_1)_x^1(x-x_0)(x-x_2)...(x-x_\kappa) + ... + (x-x_\kappa)_x^1(x-x_0)(x-x_1)...(x-x_{\kappa+1})\}] = \{поставим\ x = x_0\} =$$

$$= \frac{1}{(\kappa+1)!}\{0 + f^{(\kappa+1)}(\xi)[1(x_0-x_1)(x_0-x_2)...(x_0-x_\kappa) + 0 + ... + 0\}] = \frac{1}{(\kappa+1)!}f^{(\kappa+1)}(\xi)(-\Delta x)(-2\Delta x)*$$

$$*(-3\Delta x)(-\kappa\Delta x) = \frac{(-1)^\kappa f^{(\kappa+1)}(\xi)}{\kappa+1}(\Delta x)^\kappa$$

<u>Комментарий</u>:
При доказательстве теоремы был использован неявно тот факт, что функция ξ(x)-дифференцируемая.

<u>Замечание 4.2</u>
Погрешность усечения численного дифференцирования из теоремы 4.1 удобнее оценивать следующим образом:

$$\left|(r_\kappa(x))\,|_{x=x_0}^1\right| \le \frac{M_{\kappa+1}}{\kappa+1}(\Delta x)^\kappa, \text{ где } M_{\kappa+1} = \max_{f\in[x_0,x_\kappa]}\left| f^{(\kappa+1)}(\xi)\right| \qquad (4.10)$$

если трудно оценить значение (к+1) производной от функции f, то f заменяют на конечную разность (к+1) порядка.

$$\left|(r_\kappa(x))\,|_{x=x_0}^1\right|_{\le}^{\approx} \frac{\left|\Delta^{(\kappa+1)}y_0\right|}{(\kappa+1)\Delta x^{\kappa+1}} \qquad (4.11)$$

$$f^{(\kappa+1)}(x) \approx \frac{\Delta^{(\kappa+1)}y_0}{(\Delta x)^{\kappa+1}}$$

Погрешности усечения формул (4.6) и (4.7) оцениваются следующим образом:

Для (4.6) $\to |r_2| \le \frac{1}{6}(\Delta x)^2 M_3$

Для (4.7) $\to |r_4| \le \frac{1}{30}(\Delta x)^4 M_5$

Для (4.8) $\to |r_3| \le \frac{1}{12}h^2 M_4$

где $M_3 = \max\limits_{\xi\in[x_{-1},x_1]}\left|f^{(3)}(\xi)\right|$, $\quad M_5 = \max\limits_{\xi\in[x_{-2},x_2]}\left|f^{(5)}(\xi)\right|$, $\quad M_4 = \max\limits_{\xi\in[x_{-1},x_1]}\left|f^{(5)}(\xi)\right|$

<u>Указания:</u>
Для доказательства этих оценок необходимо продифференцировать соответственно $r_2(x)$ $r_4(x)$ в точках $x=x_1$ и x_2 соответственно. После чего сдвинуть интервалы влево $[x_0,x_2]$ сдвинуть до $\to [x_{-1},x_1]$, $[x_0,x_4] \to [x_{-2},x_2]$.

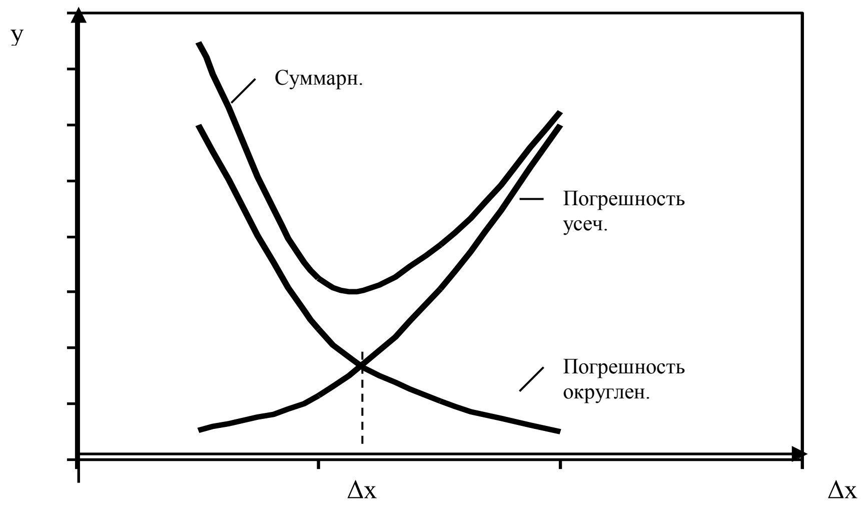
<u>Погрешности округления:</u>
Оценим погрешности округления формул (4.7) и (4.6).

$$f'(x_0) = \frac{1}{2(\Delta x)}(y_1 - y_{-1})$$

$$f'(x_0) = \frac{1}{12*2(\Delta x)}(y_{-2} - 8y_{-1} + 8y_1 - y_2)$$

Очевидно, они равны $\frac{r}{\Delta x}$ и $\frac{3}{2}\frac{r}{\Delta x}$

Погрешность усечения при уменьшении Δx уменьшается и при Δx→0 она тоже стремиться к нулю. Погрешность округления, напротив, увеличивается, а при Δx→0, она → ∞.



Шаг не следует брать слишком большой, чтобы погрешность усечения не была велика и нельзя слишком маленький, чтобы не была велика погрешность округления.

Найдем оптимальное Δx для формул (4.6) (4.7) .
Для (4.6): суммарная погрешность:

$$\frac{1}{6}(\Delta x)^2 M_3 + \frac{n}{\Delta x} \to \min$$

Находим производную по Δx и $= k\phi$

$$\frac{1}{6} * 2\Delta x * M_3 + \frac{n}{(\Delta x)^2} = 0, \ ^{-}\Delta X_{оптимальна} = \sqrt[3]{\frac{3n}{M_3}}$$

для (4.7): суммарная погрешность равна:

$$\frac{1}{30}(\Delta x)^4 * M_5 + \frac{3}{2}\frac{n}{\Delta x} \to \min$$

$$\frac{1}{30} 4(\Delta x)^3 * M_5 + \frac{3}{2}\frac{n}{(\Delta x)^2} = 0, \quad \Delta X_{опт} = \sqrt[5]{\frac{45n}{4M_5}}$$

Без вывода приведем погрешности усечения, округления формулы (4.8).
Для (4.8): погрешность усечения $\le \frac{1}{12} M_4 (\Delta x)^2$, где $M_4 = \max\left|f^{(4)}(\xi)\right|, \ \xi \in [x_{-1}, x_1]$

погрешность округления $\le \frac{4n}{(\Delta x)^2}$.

Пример (численного дифференцирования):
$f(x)=\lg' x$

x	У	Δy	$\Delta^2 y$	$\Delta^3 y$
50	1,6990	0,0414	-0,0036	0,0005
55	1,7404	0,0378	-0,0031	
60	1,7782	0,0347		
65	1,8129			

По формуле из теоремы 4:
y'(50)=1/5(0,0414-1/2(-0,0076)+1/3(0,0005))=0,00872
Погрешность усечения при этом (по теореме 4.1 и замечание 4.10) может быть оценена по формулам 4.10 и 4.11. При использовании 4.10 надо найти четвертую производную lg, а при использовании 4.11 надо найти четвертую разность от lg. Если же будем искать Δ^4(y), то потребуется еще одно значение функции и вычисление будет необходимо произвести с большей точностью (из-за быстрого уменьшения конечной разности и быстрого накапливания погрешности) ⇒ используем 4.10.

$r_3(50)\le \frac{M}{4}*5$, где $M=\max\limits_{\xi\in[50,65]}\left|\lg^{(4)}(\xi)\right|$, можно и $\left(r_3(x)\right)\Big|_{x=50}^{1}$

$\lg'(x)=\frac{1}{\ln 10*x}\ \ln'''(x)=\frac{2}{\ln 10*x^3}$

$\lg''(x)=\frac{1}{\ln 10*x^2}\ \ln^{(IV)}(x)=\frac{2}{\ln 10*x^4}$

$\max|\ln^0(x)|$ достигается при x=50 и равно $\frac{6}{\ln 10*(50)^4}\approx$ и получаем

$(r_3(x))'|_{x=50}\le\frac{2.5}{4}*10^{-4}\approx 7*10^{-5}$

<u>Таблица для погрешностей центральных формул.</u>

	$\varepsilon_{усеченное}$	$\varepsilon_{округления}$	$h_{оптимальна}$	$\varepsilon_{\min}$
4.6	$1/6h^2M_3$	η/h	$\sqrt[3]{3\eta/M_3}$	$\sqrt[3]{9\eta^2M_3/2}$
4.7	$1/30h^4M_5$	$3\eta/2h$	$\sqrt[5]{45\eta/4M_5}$	15/8 $\sqrt[5]{4\eta^4M_5/45}$
4.8	$1/12h^2M_4$	$4\eta/h^2$	$\sqrt[4]{48\eta/M_4}$	$2/\sqrt{3}\sqrt{M_4h}$

Комментарий к погрешности усеченной формулы Симпсона:

При использовании формулы Симпсона происходит интерполяция по трем узлам, а потому следует ожидать, что формула Симпсона будет точна для многочлена до второй степени включительно.

На самом деле, как мы видели из формулы (4.18), формула Симпсона будет точна для многочленов до третьей степени включительно, это явление имеет место при всех четных ***n***.

В общем случае, если n=2k+1 – нечетное, то соответствующая квадратурная формула будет точна для многочленов до степени 2k включительно, если же n=2k – четное, то соответствующая квадратурная функция будет точна до степени n+1 включительно.

§3. Численное интегрирование.

П.0. Постановка задачи.

В точках x0,x1,…,xn заданы значения функции yi=f(xi) i=0,n.

Необходимо найти значение $\int\limits_{x_0}^{x_n} f(x)dx$. Основная идея численного интегрирования: заменить функцию f(x) на интерполирующую ее функцию y(x) (которую можно проинтегрировать). $\int\limits_{x_0}^{x_n} f(x)dx \approx \int\limits_{x_0}^{x_n} y(x)dx$

Замечание:

В качестве интерполирующей функции обычно используется интерполяционный многочлен.

П.1. Квадратные формулы Ньютона- Котеса.

Как и в численном дифференцировании, в качестве интерполирующей функции будем рассматривать только интерполяционные многочлены.

Интерполяционный многочлен Лагранжа:

$(4.12) P_n(x) = \sum\limits_{i=0}^{n} g_i(x) y_i$, где

$$(4.13) g_i(x) = \frac{(x-x_0)(x-x_1)\dots(x-x_{i-1})(x-x_{i+1})\dots(x-x_n)}{(x_i-x_0)(x_i-x_1)\dots(x_i-x_{i-1})(x_i-x_{i+1})\dots(x_i-x_n)}$$

Следовательно: $\int\limits_{x_0}^{x_n} f(x)dx = \int\limits_{x_0}^{x_n} P_n(x)dx = \int\limits_{x_0}^{x_n} (\sum\limits_{i=0}^{n} g_i(x) y_i)dx = \sum\limits_{i=0}^{n} y_i \left[\int\limits_{x_0}^{x_n}\int\limits_{x_0}^{x_n} g_i(x)dx\right] = \sum\limits_{i=0}^{n} y_i A_i$ (4.14)

Коэффициенты A_i зависят от взаимного расположения узлов интерполяции x_i, где i=0,n.

Вычислим коэффициент A_i для равноотстоящих узлов интерполяции $x_i=x_0+i\Delta x$

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

$$= \begin{Bmatrix} \text{замена} \\ \text{переменных} \\ q = \dfrac{x - x_0}{\Delta x} \\ dq = \dfrac{dx}{\Delta x} \\ x = q\Delta x + x_0 \end{Bmatrix} = \int_0^n \frac{q(q-1)\ldots(q-(i-1))(q-(i+1))\ldots(q-n)}{i(i-1)\ldots-(n-i)} dq\Delta x =$$

в числителе стоит [n+1] –ая обобщенная степень q, в которой нет сомножителя (q-i), т.е. выражение $\dfrac{q^{[n]}}{q-i}$

$$= \frac{(-1)^{n-i}\Delta x}{i!(n-i)!}\int_0^n \frac{q^{[n]}}{q-i} dq = ((b = x_n) - (a = x_0))H_i \text{, где}$$

$$H_i = \frac{(-1)^{n-i}}{i!(n-i)!\,n}\int_0^n \frac{q^{[n]}}{q-i} dq \quad \text{i=0,n} \qquad (4.15)$$

Коэффициенты Ньютона - Котеса зависят от n и от i и не зависят от Δx.

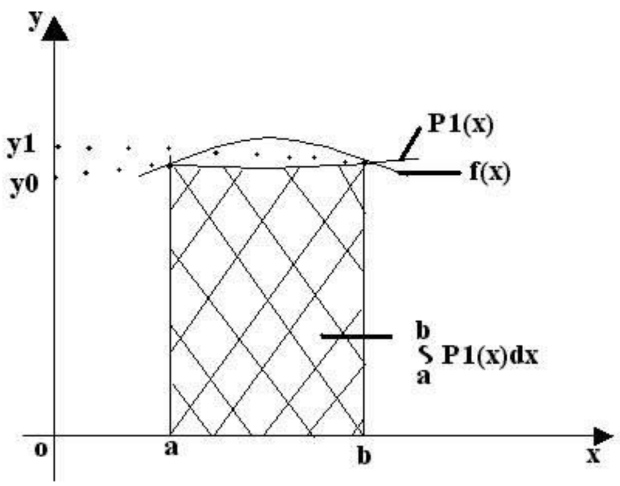
П.2. Формула трапеции. Формула Симсона.

Вычислим коэффициент Ньютона- Котеса для n=1.

Имеем $H_0 = \dfrac{(-1)^{1-0}}{0!*1!*1}\displaystyle\int_0^1 \frac{q(q-1)}{q} dq = (-1)\int_0^1 (q-1)dq = -(\frac{q}{2}^2 - q)_0^1 |= \frac{1}{2} - 0 = \frac{1}{2}$

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

Следовательно: $\displaystyle\int_{a=x_0}^{b=x_1} f(x)dx \approx A_0 y_0 + A_1 y_1 = \frac{b-a}{2} y_0 + \frac{b-a}{2} y_1 = (b-a)\left(\frac{y_0 + y_1}{2}\right)$



$$\int_a^b f(x)dx \approx \int_a^b P_1(x)dx = S_{трапеции}$$

Вычислим коэффициенты Н.-К. для n=2.

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

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

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

По формуле (4.15) получаем:

$$\int_{x_0}^{x_2} f(x)dx = A_0y_0 + A_1y_1 + A_2y_2 = (b-a)\left(\frac{y_0}{6} + \frac{2}{3}y_1 + \frac{1}{6}y_2\right)$$ -формула Симпсона.

Таблица коэффициентов Котеса.

i \ n	1	2	3	4	5	6	7	8
H_0	½	1/6	1/8	7/90	19/288	41/840	751/17280	989/28350
H_1	½	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

П.3. Погрешность формул численного интегрирования.

При численном интегрировании возникают два типа погрешностей:

1. усечения
2. округления.

Погрешность усечения возникает из-за замены функции f(x) на интерполирующие ее многочлены. Погрешность округления возникает из-за того, что значение функции y_i в узлах интерполяции известно не точно, а приближенно, с некоторой погрешностью η. Оценим погрешность формулы усечения трапеции:

$$R=\int_{x_0}^{x_1} f(x)dx-\frac{x_1-x_0}{2}(f(x_1)+f(x_0))$$

Рассмотрим погрешность усечения R, как функцию шага ΔX при этом $x_1=x_0+\Delta x$

$$R=\int_{x_0}^{x_1+\Delta x} f(x)dx-\frac{\Delta x}{2}(f(x_0+\Delta x)+f(x_0))$$

$$F(x)=\int_{a(x)}^{b(x)} f(t)dt$$

$$F'(x)=f(b(x))*b'(x)-f(a(x))*a'(x)$$

$$R'(\Delta x)=f(x_0+\Delta x)*1-f(x_0)*0-\frac{1}{2}\left[1*\left(f(x_0+\Delta x)+f(x_0)+\Delta x(f'(x_0+\Delta x)\right)\right]=$$

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

R'(0)=0 (так же как и R(0)=0).

Найдем $R''(\Delta x)=(f(x_0+\Delta x)*1-f(x_0)*0-\frac{1}{2}[1*(f(x_0+\Delta x)+f(x_0)+\Delta x(f'(x_0+\Delta x))])'=$

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

Проинтегрируем дважды по Δx учитывая, что R'(0)=R(0)=0.

Используем теорему о среднем (из курса высшей математики):

$\int_a^b f(x)dx=(b-a)*f(\xi)$, где $\xi\in(a,b)$.

<u>Другой вариант теоремы о среднем.</u>

$$\int_a^b f(g(t))h(t)dt=f(g(\xi))\int_a^b h(t)dt$$

Для любой непрерывной функции f, g и h найдется точка ξ такая, что имеет место равенство:

интегрируем первый раз R'(h) имеем

$$R'(h)=\int_0^h R''(s)ds+R'(0)=(*)$$

$$R(h)=\int_{x_0}^{x_0+h} f(t)dt-\frac{h}{2}\left[f(x_0+L)+f(x_0)\right]$$

$$(*)=\int_0^h -\frac{h(t)}{2}f''(x_0+t)dt=\begin{Bmatrix}теорема-о\\ среднем\end{Bmatrix}=f''(x_0+\xi)\int_0^h -\frac{t}{2}dt=-f''(x_0+\xi(h))\frac{h^2}{4}$$

$$R(h)=\int_0^h R'(s)ds+R(0)=\int_0^h -f''(x_0+\xi(s))\frac{s^2}{4}ds=-f''(x_0+\xi(\lambda))\int_0^h\frac{s^2}{4}ds=-f''(x_0+\xi(\lambda))\frac{h^3}{12}$$

$$R''(h)=-\frac{h}{2}f''(x_0+h)\qquad(4.16)$$

$R(h)=-\frac{h^3}{12}f''(\bar{x})$, где $\bar{x}\in[x_0,x_0+h]$ (4.17)

Формула трапеции верна для линейных функций.

Если f’’ всюду > 0, т.е. функция выпукла вниз, то интеграл по формуле трапеции найден с избытком (R<0).
Если f’’<0, то R>0 и интеграл по формуле трапеции найден с недостатком.
Оценка погрешности усечения формулы Симпсона:
Узлы x_0, x_0+h; x_0+2h переведем в узлы x_0-h, x_0, x_0+h и формула Симпсона для узлов примет вид:

$$\int_{x_0-h}^{x_0+h} f(x)dx \approx 2h\left[\frac{1}{6}f(x_0-h)+\frac{2}{3}f(x_0)+\frac{1}{6}f(x_0+h)\right]$$

$$R(h)=\int_{x_0-h}^{x_0+h} f(x)dx-\frac{h}{3}f(x_0-h)-\frac{4}{3}f(x_0)-\frac{h}{3}f(x_0+h)$$

$$R'(h)=f(x_0-h)*1-f(x_0-h)*(-1)-\frac{1}{3}f(x_0-h)+\frac{h}{3}f'(x_0-h)-\frac{4}{3}f(x_0)-\frac{1}{3}f(x_0+h)-\frac{h}{3}f'(x_0+h)$$

$$R(0)=0; R'(0)=0$$

$$R''(h)=f'(x_0+h)-f'(x_0-h)+\frac{1}{3}f'(x_0-h)+\frac{1}{3}f'(x_0-h)-\frac{4}{3}f(x_0)-\frac{1}{3}f(x_0+h)-\frac{h}{3}f'(x_0+h)$$

$$R''(h)=f'(x_0+h)-f'(x_0-h)+\frac{1}{3}f'(x_0-h)+\frac{1}{3}f'(x_0-h)-\frac{h}{3}f''(x_0-h)-\frac{1}{3}f'(x_0+h)-$$

$$-\frac{1}{3}f(x_0+h)-\frac{h}{3}f''(x_0+h)=\frac{1}{3}f'(x_0+h)-\frac{1}{3}f'(x_0-h)-\frac{h}{3}\left[f''(x_0+h)+f''(x_0-h)\right]$$

$$R'''(h)=-\frac{2h^2}{3}f^{(IV)}(\xi)$$

Трижды проинтегрируем R"(h)

$$R''(h)=\int_0^h R'''(s)ds+R''(0)=-f^{(IV)}(\lambda)\int_0^h\frac{2s^2}{3}ds=-f^{(IV)}(\lambda)\frac{2}{9}h^3$$

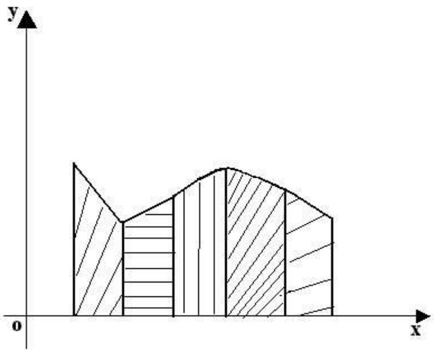
$$R'(h)=\int_0^h R''(s)ds+R'(0)=-f^{(IV)}(\Theta)\frac{1}{18}h^4$$

$$R(h)=\int_0^h R'(s)ds+R(0)=-f^{(IV)}(\bar{x})\frac{1}{90}h^5, где\ \bar{x}\in[x_0-h,x_0+h]$$

$$R(h)==-f^{(IV)}(\bar{x})\frac{1}{90}h^5, где\ \bar{x}\in[x_0-h,x_0+h] \qquad (4.18)$$

Комментарий к погрешности усечения формулы Симпсона:
Формула точна для многочленов до третьей степени включительно. При использовании формулы Симпсона происходит интерполяция по трем узлам, поэтому формула Симпсона точна для многочленов до второй степени включительно. На самом деле, как видно из выше рассмотренной формулы она будет точна для многочлена третей степени –парадокс: при n=7 – до многочлена седьмой степени, при n=8 до многочлена восьмой степени. В общем случае, если n=2k+1 – нечетное, то соответствующая квадратурная формула будет точна для многочлена до степени 2k включительно. Если n=2k – четное, то квадратурная формула будет точна для многочлена до степени 2k+1 включительно.
Квадратурные формулы (формулы вида 4.14) более высоких степеней

практически не используется, если необходимо вычислить $\int\limits_{x_0}^{x_n} f(x)dx$ при большом числе узлов интерполяции (n), то весь интервал (x_0,x_n) разбивают на маленькие участки интерполирования. Для формулы трапеции две точки – один интервал, для формулы Симпсона три точки – два интервала.



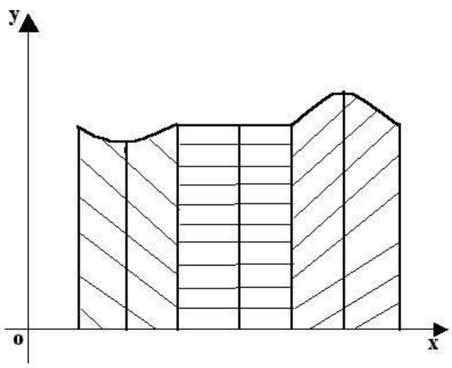
Общий график есть сумма графиков по коротким отрезкам.
Общая формула трапеции:

$$\int\limits_{a=x_0}^{b=x_n} f(x)dx = \int\limits_{x_0}^{x_1} f(x)dx + \int\limits_{x_1}^{x_2} f(x)dx + ... + \int\limits_{x_{n-1}}^{x_n} f(x)dx = \left\{\begin{matrix} на \\ кажом \\ интервале \\ (x_{i-1}, x_i) \end{matrix}\right\} =$$

$$\overset{\approx}{=} \Delta x\left[\frac{y_0 + y_1}{2}\right] + \Delta x\left[\frac{y_1 + y_2}{2}\right] + ... + \Delta x\left[\frac{y_{n-1} + y_n}{2}\right] = \frac{b-a}{n}\sum_{i=1}^{n}\frac{y_{i-1} + y_i}{2} = \qquad (4.19)$$

$$= (\frac{b-a}{n} = \Delta x)\left[\frac{1}{2}y_0 + \frac{1}{2}y_n + \sum_{i=1}^{n-1} y_i\right] \approx \int\limits_{a=x_0}^{b=x_n} f(x)dx$$

Общая формула Симпсона:



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

$$\overset{\approx}{=} 2(\Delta x = x_2 - x_0)\left[\frac{1}{6}y_0 + \frac{2}{3}y_1 + \frac{1}{6}y_2\right] + 2(\Delta x = x_4 - x_2)\left[\frac{1}{6}y_2 + \frac{2}{3}y_3 + \frac{1}{6}y_4\right] + ... + \quad (4.20)$$

$$+ 2\Delta x\left[\frac{1}{6}y_{n-2} + \frac{2}{3}y_{n-1} + \frac{1}{6}y_n\right] = \frac{b-a}{k}\sum_{i=1}^{k}\left[\frac{1}{6}y_{2i-2} + \frac{2}{3}y_{2i-1} + \frac{1}{6}y_{2i}\right] =$$

$$= \frac{b-a}{k}\left[\frac{1}{6}y_0 + \frac{1}{6}y_n + \frac{2}{3}(y_1 + y_3 + y_5 + ... + y_{2n-1}) + \frac{1}{3}(y_2 + y_4 + ... + y_{n-2})\right] \approx \int_{x_0}^{x_n} f(x)dx$$

Погрешности усечения общей формулы трапеции и общей формулы Симпсона состоят из суммы погрешностей усечения формулы трапеции и формулы Симпсона на каждом интервале.
Для формулы трапеции погрешность усечения R:

$$R = (R_1 \in [x_0, x_1]) + (R_2 \in [x_1, x_2]) + ... + R_n = \{(4.17)\} = -\frac{h^3}{12}f''(\bar{x}_1 = [x_0, x_1]) - \frac{h^3}{12}f''(\bar{x}_2 = [x_1, x_2]) + \quad (4.21)$$

$$+ ... + \frac{h^3}{12}f''(\bar{x}_n = [x_{n-1}, x_n]) = (n = \frac{b-a}{h})(-\frac{h^3}{12})\sum_{i=1}^{n} f''(\bar{x}_i) * \frac{1}{n} = -(b-a)\frac{h^2}{12}f''(\hat{x}) =$$

$= R_{\text{усечения для общей формулы трапеции}}$.
Аналогично выводу формулы (4.21):

$R_{\text{усечения для общей формулы Симпсона}} = -(b-a)\frac{h^4}{180}f^{(IV)}(\hat{x})$ (4.22)

Погрешность округления общей формулы трапеции и общей формулы Симпсона: $(b-a)*\eta$.

Общая формула Симпсона: $\frac{h}{3}\sum_{i=1}^{k}(y_{2i-2} + y_{2i-1} + y_{2i})$,

погрешность округления $\le (b-a)*\eta$.

При использовании квадратурных формул до порядка $n \le 7$ погрешность округления не превосходит $(b-a)*\eta*\sum_{i=1}^{n}|H_i|$, с $n>7$ $\sum_{i=1}^{n}|H_i|>1$ растет очень быстро.

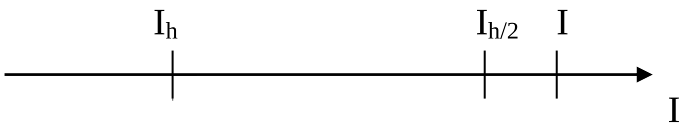
П.4. Метод двойного пересчета.

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

Идея метода:

Вычислить интеграл по формуле трапеции с шагом h-I_h, и вычислить интеграл по формуле трапеции с шагом h/2-$I_{h/2}$.При этом по формуле (4.21) имеем: $I - I_h = \frac{(b-a)}{12}f''(\bar{x}) * h^2$, аналогично $I - I_{h/2} = -\frac{(b-a)}{12}f''(\bar{\bar{x}}) * \left(\frac{h}{2}\right)^2$.

Предположим, что: $f''(\bar{x}) = f''(\bar{\bar{x}})$, в этом случае: $I-I_h=4(I-I_{h/2})$, (т.е. верхняя величина в 4 раза меньше нижней).



Поэтому: $I_h - I_{h/2} = 3(I_{h/2} - I)$.
Следовательно, если $I_h - I_{h/2} < 3\varepsilon$, то $|I_h - I_{h/2}| < \varepsilon$.
Вывод (практический):
Вычислить I_h, $I_{h/2}$, если их разность по модулю не превосходит 3ε, где ε - заданная погрешность интеграла, то считают, что: $I_{h/2}$ – значение интеграла с погрешностью ε.

Коррекция после двойного пересчета:

Как видно из графика, в качестве $I_{\text{точного}}$ выгодно использовать: $I = I_{h/2} + 1/3(I_{h/2} - I_h) = I_{\text{кор}}$ После коррекции по этой формуле точность метода возрастает на порядок, т.е. метод трапеции будет не второго, а третьего; а Симпсона – пятого порядка точности. $|I_h - I_{h/2}| < 3\varepsilon$.
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 5: Решение дифференциальных уравнений и систем дифференциальных уравнений.
П.0. Постановка задачи.

Дифференциальное уравнение – это уравнение, в котором участвует $f(x,y,y'\ldots y^{(k-1)})=0$ – ДУ $k^{\text{того}}$ порядка.
Обычно ДУ бывает разрешено относительно старшей производной, т.е.

$y^{(k)} = f(x,y,y'\ldots y^{(k-1)})$

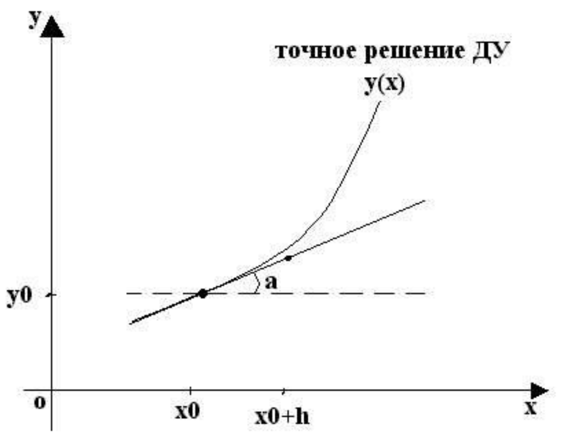
Необходимо решить дифференциальное уравнения или систему дифференциальных уравнений, т.е. найти решение этого уравнения или системы в любой заданной точке с заданной точностью.

П.1. Простейший вариант задачи (уравнение первого порядка с задачей Коши) и простейшие методы решения (Эйлера, Рунге- Кутта).

Задача Коши:

$$\begin{cases} y'(x) = f(x,y) \\ y(x_0) = y_0 \end{cases} \qquad (5.1)$$

Общая идея всех методов численного решения ДУ и СДУ:
Зафиксируем шаг h. Найдем последовательно с помощью некоторого стандартного метода $y(x_0+h), y(x_0+2h), \ldots, y(x_0+nh)=$, пока не попадем в интересующую нас точку
$=y(x_n)=y(b)$, где b – заданная точка. Шаг h выбирается достаточно маленьким, чтобы полученное значение y(b) отличалось от точного решения ДУ не больше, чем на заданную погрешность ε.
Метод Эйлера:



$tg\alpha = y'(x_0) = \{(5.1)\} = f(x_0, y_0)$
В методе Эйлера заменяют точное значение ($y(x_0+h)$) на значение касательной, проведенной к графику y(x) в точке (x_0,y_0), т.е. $y(x_0+h) \approx y(x_0) + hf(x_0, y_0)$ (5.2)

Аналогично: $y(x_2) \approx y(x_1) + hf(x_1, y_1)$ - формула метода Эйлера

<u>Обоснование метода Эйлера и его модификации. Оценка локальной погрешности этих методов.</u>

$$y(x_0+h) = y(x_0) + \int_0^h y'(x_0+t)dt \qquad (5.3)$$

Если заменить $\int_0^h y'(x_0+t)dt$ по формуле прямоугольника на $hy'(x_0)=hf(x_0,y_0)$, то формула (5.3) преобразуется в формулу (5.2). При этом локальная погрешность формулы Эйлера:

$\int_0^h y'(x_0+f)dt - hy'(x_0) = const * h^2$. Если в формуле (5.3) заменить интеграл $\int_0^h y'(x_0+t)dt$, по формуле трапеции, на $\frac{h}{2}(y'(x_0) + y'(x_0+h))$, то локальная погрешность = $const*h^3$ (5.4)

$f(x_0,y_0)$ (по формуле 5.1) $f(x_0+h, y(x_0+h)$, где x_0 найдем по формуле Эйлера.

Получаем набор формул:

$$y(x_0+h) = y(x_0) + \frac{h}{2}\left[f(x_0, y_0) + f(x_0+h, y_1^*)\right], \text{ где } y_1^* = y(x_0) + hf(x_0, y_0) \qquad (5.5)$$

Оценим погрешность формулы (5.5)
используя (5.2) получаем, что погрешность равна:

$$\frac{h}{2}\left[f(x_0,y_0)+f(x_0+h,y_1^*)\right]-\int_0^h y'(x_0+t)dt=\frac{h}{2}\left[f(x_0,y_0)+f(x_0+h,y(x_0+h))\right]-\int_0^h y'(x_0+t)dt-$$

$$-\frac{h}{2}\left[f(x_0,y_0)+f(x_0+h,y(x_0+h))\right]+\frac{h}{2}\left[f(x_0,y_0)+f(x_0+h,y_1^*)\right]=\{(5.4)\}=$$

$$=consth^3+\frac{h}{2}\left[f(x_0+h,y_1^*)-f(x_0+h,y(x_0+h))\right]=consth^3+\frac{h}{2}f'\,y(x_0+h,\hat{y})consth^2=consth^3$$

Каждый раз const может быть разной.
Формулы (5.5) (с локальной погрешностью const*h^3), называются модификацией метода Эйлера или методом Рунге- Кутта – 2-ого порядка.
<u>Вторая модификация метода Эйлера (метод Рунге- Кутта 2-ого порядка):</u>

Если $\int_0^n y'(x_0+t)dt$ на hy’(x$_0$+h/2) (с погрешностью const*h^3)

(5.6) $y(x_0+h)=y(x_0)+h*f(x_0+\frac{h}{2};y^*_{\frac{1}{2}})$, где $y^*_{\frac{1}{2}}=y(x_0)+\frac{h}{2}f(x_0,y_0)=$

локальная погрешность формулы (5.6)-const*h^3.

П.2. **<u>Сведение дифференциальных уравнений высших порядков к системе дифференциальных уравнений первого порядка.</u>**

Рассмотрим дифференциальное уравнение n$^{\underline{ого}}$ порядка, разрешенное относительно старшей производной.
Задачи Коши:
(5.7) $y^{(n)}(x)=f(x,y,y',y'',...,y^{(n-1)})$ дифференциальное уравнение

$\begin{cases} y(x_0)=y_0^* \\ y'(x_0)=y_0^{'} \\ y^{(n-1)}(x_0)=y_0^{n-1} \end{cases}$ - начальные условия.

Чтобы свести (5.7) к СДУ 1-ого порядка

$Y(x)=\begin{pmatrix} y(x) \\ y'(x) \\ y''(x) \\ : \\ y^{(n-1)}(x) \end{pmatrix}$,тогда: $Y'(x)=\begin{pmatrix} y'(x) \\ y''(x) \\ : \\ y^{(n-1)}(x) \\ y^{(n)}(x) \end{pmatrix}=\begin{pmatrix} y'(x) \\ y''(x) \\ : \\ y^{(n-1)}(x) \\ f(x,y,y',...,y^{(n-1)}) \end{pmatrix}=F(X,(Y=y,y',...,y^{(n-1)}))$

$Y(x_0)=\begin{pmatrix} y_0 \\ y'_0 \\ y''_0 \\ : \\ y_0{}^{(n-1)} \end{pmatrix}$ (5.8)

Таким образом (5.7) преобразовалось к $\begin{cases} Y'=F(X,Y) \\ Y(x_0)=Y_0 \end{cases}$ - СДУ в векторной форме.

Решить СДУ (5.8) можно любым известным способом.

Сведение ДУ высших порядков к СДУ первого порядка.

Рассмотрим ДУ n^ого^ порядка.

$$\begin{cases} y^{(n)} = f(x, y, y'...y^{(n-1)}) \\ \begin{cases} y(x_0) = y_0^{(0)} \\ y^{(n-1)}(x_0) = y_0^{(n-1)} \end{cases} \end{cases} \qquad (5.8.1.)$$

Чтобы свести ДУ (5.8.1.) n^го^ порядка к СДУ первого порядка, введем вектор - функцию Y(x), тогда для этой векторной функции справедливо следующее ДУ.

$$Y(x) = \begin{pmatrix} y \\ y' \\ \\ y^{(n-1)} \end{pmatrix} \qquad Y'(x) = \begin{pmatrix} y' \\ y'' \\ \\ y^{(n-1)} \\ f(x), y'...y^{(n-1)} \end{pmatrix}$$

В итоге ДУ (5.8.1.) преобразуется к СДУ (5.8.2.):

$$\begin{cases} Y' = F(x, Y) \\ Y(x_0) = Y_0 \end{cases} \quad (5.8.2.), \text{ где}$$

$$F(x, y, y'...y^{(n-1)}) = \begin{pmatrix} y' \\ y'' \\ \\ y^{(n-1)} \\ f(x, y...y^{(n-1)}) \end{pmatrix}, \qquad Y_0 = \begin{pmatrix} y_0^{(0)} \\ \\ y_0^{(n-1)} \end{pmatrix}$$

(5.8.2.) можно решать методом Эйлера и Рунге-Кутта, только работать не с числами, а с векторами.

Сведение дифференциальных уравнений 2-ого порядка к СДУ первого порядка и выполнение 2-х шагов:

$$\begin{cases} y'' - 2y'x + y = 1 \\ y(1) = 2 \\ y'(1) = -2 \end{cases}$$, вводим $Y = \begin{pmatrix} y \\ y' \end{pmatrix}$. Имеем для Y ДУ(СДУ)

$$\begin{cases} Y' = \begin{pmatrix} y' \\ 1 + 2y'x - y \end{pmatrix} \\ Y(1) = \begin{pmatrix} 2 \\ -2 \end{pmatrix} \end{cases}$$,возьмем шаг h=0,1. Имеем:

$$Y(1{,}1) = Y(1) + 0{,}1F(1, Y(1)) = \begin{pmatrix} 2 \\ -2 \end{pmatrix} + 0{,}1\begin{pmatrix} -2 \\ 1 + 2*(-2)*1 - 2 \end{pmatrix} = \begin{pmatrix} 1{,}8 \\ -2{,}5 \end{pmatrix} \begin{matrix} \approx y(1{,}1) \\ \approx y'(1{,}1) \end{matrix}$$

$$Y(1{,}2) = Y(1{,}1) + 0{,}1F(1{,}1, Y(1{,}1)) = \begin{pmatrix} 1.8 \\ -2{,}5 \end{pmatrix} + 0{,}1\begin{pmatrix} -2{,}5 \\ 1 + 2(-2{,}5)1{,}1 - 1{,}8 \end{pmatrix} = \begin{pmatrix} 1{,}55 \\ -3{,}13 \end{pmatrix} \begin{matrix} \approx y(1{,}2) \\ \approx y'(1{,}2) \end{matrix}$$

П.3. Метод Рунге- Кутта 4-ого порядка.

$$k_1 = f(x_i, y_i)$$
$$k_2 = f(x_i + \frac{h}{2}, y_i + \frac{h}{2} k_1)$$
$$k_3 = f(x_i + \frac{h}{2}, y_i + \frac{h}{2} k_2) \qquad (5.9)$$
$$k_4 = f(x_i + h, y_i + h k_3)$$

$y_{i+1} = y_i + \frac{h}{6}(k_1 + 2k_2 + 2k_3 + k_4)$ - формулы Р.-К. 4-ого порядка.

Погрешность на одном шаге метода Рунге- Кутта =const h^5 – локальная погрешность. Глобальная погрешность (т.е. погрешность вычисляется на фиксированном интервале [a,b]) всегда на порядок меньше локальной погрешности.

Методы	Локальная	Глобальная
М.Э. – (5.2)	const*h^2	const*h
М.Р. – К. 2пор(1) – (5.5)	const*h^3	const*h^2
М.Р. – К. 2пор(2) – (5.6)	const*h^3	const*h^2
М.Р. – К. – (5.9)	const*h^5	const*h^4

<u>П.4. Оценка погрешности методов Эйлера и Рунге- Кутта.</u>

<u>Лемма 5.1</u>:

Решения задачи Коши: $\begin{cases} y'(x) = y(x) f(x) \\ y(a) = y_0 \end{cases}$ (5.10)

Будет функция: $y(x) = y_0 * \exp(\int_a^x f(t) dt)$ (5.11)

$$\frac{dy}{dx} = y f(x)$$
$$\frac{dy}{y} = f(x) dx$$
$$\ln y = \int \frac{dy}{y} = \int f(x) dx + c_1$$
$$F(x) - F(a) = \int_a^x f(z) dt$$
$$y(x) = \exp\left(\int_a^x f(z) dt\right) * c$$

Используя условие y(a)=y_0 получаем $y(a) = \exp(\int_a^x f(z) dt) * c \Rightarrow c = y_0$

<u>Лемма 5.2:</u>

Пусть функции $y_1(x)$, $y_2(x)$ являются решениями задачи Коши с одной и той же функцией f и начальными условиями, заданными в одной точке, т.е.

$y_1(x)$ $\begin{cases} y_1'(x) = f(x, y_1) \\ y_1(a) = y_1 \end{cases}$

$y_2(x)$ $\begin{cases} y_2'(x) = f(x, y_2) \\ y_2(a) = y_2 \end{cases}$

тогда существует функция $\bar{y}(x) \in [y_1(x), y_2(x)]$ - такая, что для любой точки b имеет место равенство $y_2(b) - y_1(b) = (y_2(a) - y_1(a)) * \exp(\int_a^b f_y'(x, \bar{y}(x))dx)$ (5.11.1)

Доказательство:

$\forall x \in [a,b]$ имеем: $(y_2(x) - y_1(x))' = f(x, y_2(x)) - f(x, y_1(x)) = f_y'(x, \bar{y}(x)) * [y_2(x) - y_1(x)]$

Воспользуемся леммой 5.1, взяв $y=y_2-y_1$, $f=f'y$ получаем нужное нам утверждение. Как следует из формулы (5.11.1) расхождение между решениями задачи Коши с одним и тем же ДУ но разными начальными условиями, (заданными в одной точке) оцениваются через разность начальных условий и экспоненты от $\int f_y'(x, \bar{y}(x))dx$.

Теорема 5.3: (оценка погрешности метода Эйлера и Рунге- Кутта)

Рассмотрим любой метод, имеющий локальную погрешность const h^{p+1}, тогда на интервале (a,b) решение дифференциального уравнения (5.1), полученное по этому методу отличается от точного решения не больше, чем на (5.12) $\exp(L(b-a))[consth^p(b-a) + n\delta + \Delta]$, где $L = \max_{(x,y)\in D} |f_y'(x,y)|$, δ-max погрешность на каждом шаге, Δ- погрешность, с которой было известно y_0, n – число шагов =(b-a)/h.

Комментарий к теореме (5.3) и формуле (5.12):

Если при $h\to 0$, $\Delta\to 0$, $n\delta \to 0$, то погрешность решения дифференциального уравнения стремится к нулю. Но на практике, если (exp(L(b-a))) - велико, то достичь малой погрешности бывает практически невозможно.

Теорема 5.4(оценка глобальных погрешностей одношаговых методов)

Если в области D: $f'y(x,y)\le -k<0$, где k – некоторая const, то оценку теоремы 5.3. можно быть заменить на оценку $|y_n - y(x_n)| \le const(h^p + \frac{\delta}{nk}) + \Delta \cdot \exp(-k(b-a))$

Доказательство:

$$|y_n - y(x_n)| \le (*) \le (consth^{p+1} + \delta) + (consth^{p+1} + \delta)\exp(-kh) + (consth^{p+1} + \delta)\exp(-krh) + ... +$$

$$+ (consth^{p+1} + \delta)\exp(-k(n-1)h) + const(h^{p+1} + \delta)\exp(-knh) \approx \frac{const(h^{p+1} + \delta)}{kh} + \Delta\exp(-knh) =$$

$$= \left(\frac{const}{k} h^p + \frac{\delta}{kn}\right) + \Delta\exp(-knh) ----(5.13) \le const(h^p + \frac{\delta}{hk}) + \Delta\exp(-k(b-a))$$

Комментарий к формуле 5.13:

В 5.13 нет большого множителя exp(L(b-a)), поэтому оценка в 5.13 гораздо меньше, а, следовательно, лучше. Но чтобы 5.13 была верна, надо потребовать $f'y \le -k < 0$. Область D - эта область на плоскости XOY, на которой все происходит.

п 5. Многошаговые методы решения ДУ и СДУ

Рассмотренные выше методы модификаций Эйлера, Рунге-Кутта – одношаговые методы, т.к. в каждом из них для нахождения Y_{i+1} использовалось только одно значение $У_i$, найденное на предыдущем шаге. В многошаговых методах используется для нахождения $У_{i+1}$ не только $У_i$, но и $У_{i-1}$, $У_{i-2}$ и т.д. Метод называется n шаговым, если для нахождения $У_{i+1}$ использовалось n предыдущих значений. Многошаговые методы дают лучший результат по сравнению с одношаговыми. Они менее чувствительны к ошибкам вычислений, погрешность у них накапливается медленнее.

Многошаговых методов много. С повышением точности на ЭВМ актуальность многошаговых методов снизилась. Наиболее распространенным является метод Милна. Сначала находим $y_i = y_{i-4} + \frac{4h}{3}(2f(x_{i-3}, y_{i-3}) - f(x_{i-2}, y_{i-2}) + 2f(x_{i-1}, y_{i-1}))$

Затем $y_i = y_{i-2} + \frac{h}{3}(f(x_{i-2}, y_{i-2}) + 4f(x_{i-1}, y_{i-1}) + f(x_i, y_i))$ Метод Милна 4-х шаговый метод (для нахождения y_{i+1} используются четыре предыдущих значение).

Погрешность (глобальная) метода Милна макая же, как у метода Рунге-Кутта четвертого порядка: const*n^4.

п 6. Оценка погрешности решения ДУ и СДУ методом двойного пересчета. Коррекция решения.

Явные оценки для погрешности решения весьма затруднительны, т.к. const, участвующие в оценке погрешности весьма трудно оценить, поэтому поступают, как и в численном интегрировании – используют двойной пересчет. Т.е. находят решение ДУ на [a,b] дважды с шагом h и с шагом h/2. после этого сравнивают значения во всех точках x (в которых были вычислены оба значения) и их разность с 1ε для первого порядка, 3ε для второго порядка, 15ε для четвертого порядка.

Замечание 1: если ε не достигнута, т.е. хотя бы в одной точке неравенство не выполняется, то h делят на 2 и т.д. 2: после двойного пересчета необходимо сделать коррекцию, т.е. где были вычислены значения с шагом h и h/2 можно найти $y_{кор}$

$$y_{кор} = y_{h/2} + \frac{1}{2^p + 1}(y_{h/2} - y_h) \qquad (5.13.1)$$

В качестве точного значения можно взять $y_{h/2}$ или $y_{кор.}$

$y_{кор.}$- точнее, после коррекции точность метода возрастает на порядок.

3: если требуется найти решение ДУ в точке x, которая не совпадает ни с одним из узлов x_n, используют интерполяцию.

4: при двойном пересчете и расчете СДУ или ДУ высших порядков, сведенных к СДУ, мы можем контролировать выполнение неравенства (5.13.1) не только для у, но и для у',у''.
5: где и как можно применять метод двойного пересчета?
При решении ДУ и СДУ, при численном интегрировании, очевидно, что этот метод можно применять к задачам численного дифференцирования (так как меняется шаг и аналогичная погрешность).

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. ДУ + начальные условия, которые заданы в разных точках.
Рассмотрим простейшую краевую задачу для ДУ 2 порядка.

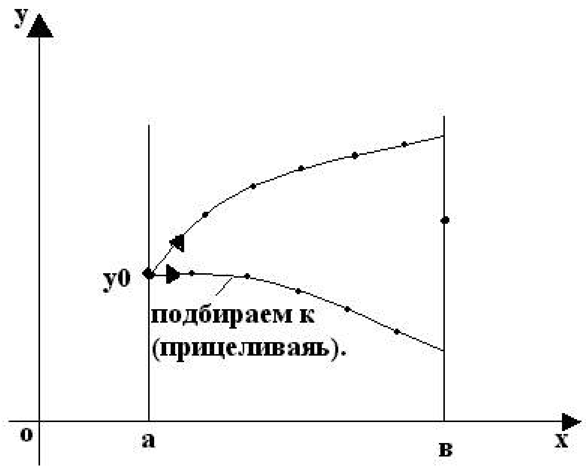
$$\begin{cases} y'' = f(x, y, y') \\ y(a) = y_0 \\ y(b) = y_1 \end{cases} \qquad (5.14)$$

Для решения 5.14 решают набор обычных задач Коши.

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

(5.15), где параметр k меняют, подбирая его, чтобы выполнялось условие у(в) = $у_1$

Этот метод называется методом стрельб



Процесс перестрелки ведем до тех пор, пока не будет достигнута заданная точность. Задачу Коши (5.15) решаем любым методом: Эйлера, Коши, Милна, Рунге-Кутта. При прицеливании необходимо решить нелинейное уравнение у(к)=$у_1$, где у(к) – решение задачи 5.15 в точке х=в. Оно зависит от числа *k,* которое можно найти с некоторой заданной точностью.

Рекомендации:

Так как ДУ не может быть решено относительно старшей производной $y^{(n)}$ то на каждом шаге выбранного метода необходимо находить старшую производную из ДУ т.е. на каждом шаге необходимо решать нелинейное уравнение для старшей производной. При этом выгодно использовать самый быстрый способ решения НУ (м.Ньютона), который может не сходиться, если первоначальное приближение x_0 далеко от корня, поэтому до использования метода Ньютона необходимо сделать несколько шагов МПД, который всегда гарантировано сходится.

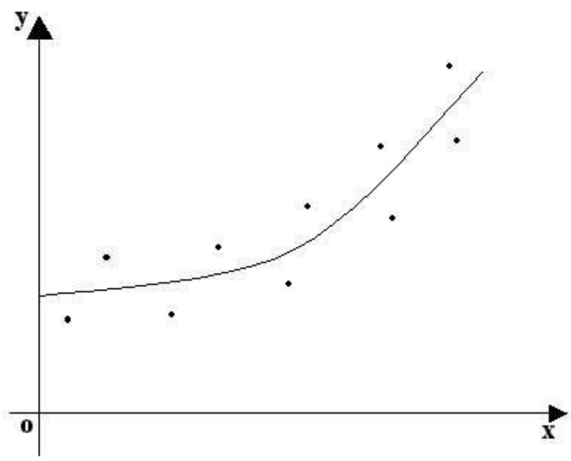
Замечание к теме 5:

Во всех формулах (Эйлера, Милна, Рунге-Кутта) решение ДУ 1-ого порядка можно заменить $y \to У$, $f \to F$ и решать, таким образом, не ДУ 1-ого порядка, а СДУ 1-ого порядка.

Тема 6: Аппроксимация.

П.0. Постановка задачи аппроксимации.

Функция $\mathbf{y=f(x)}$ задана своими значениями в точках $\mathbf{x_0,x_1,...,x_n}$, $\mathbf{y_i=f(x_i)}$.
Фиксирован некоторый класс функций **G**.



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

Уточнение задачи аппроксимации.

Для оценки близости функции **g** к набору $\mathbf{y_i}$ составляется вектор невязок (погрешности) $\mathbf{(g(x_0)-y_0, g(x_1)-y_1, \dots , g(x_n)-y_n)}$. Его надо минимизировать с помощью выбора функции **g**. В аппроксимации по методу наименьших квадратов используется 2-я норма. В классе функций **G** рассматривают всевозможные линейные комбинации.$\mathbf{G=\{a_0g_0(x)+a_1g_1(x)+...+a_kg_k(x),a_iR\}}$, $\mathbf{g_i(x)}$ некоторый фиксированный набор функции.

Пример: $G=\{a_0\}$ аппроксимация константы

$G=\{a_0+a_1x\}$

$G=\{a_0+a_1x+a_2x\}$

П.1. Метод минимальных квадратов (ММК).

Задача: $\|R_2\| \to \min$ за счет выбора коэффициентов a_j в аппроксимирующей функции $g(x) = \sum_{j=0}^{k} a_j g_j(x)$.

Фиксирован набор функций: $\mathbf{g_0(x), g_1(x), \ldots, g_k(x)}$ даны $\mathbf{(x_0,y_0),(x_1,y_1),\ldots,(x_n,y_n)}$.
Задача:

g(x)= $\sum_{j=0}^{k} a_j g_j(x)$ - линейная комбинация функции $\mathbf{g_j}$

составляем вектор невязок: $(g(x_0)-y_0,\ g(x_1)-y_1,\ \ldots,\ g(x_n)-y_n)$ и его 2-ю норму, стремящуюся к минимальному (изменяя aj)

$$S(a_0 \ldots a_k) = (\| (g(x_0) - y_0, g(x_1) - y_1, \ldots, g(x_n) - y_n) \|_2)^2 =$$
$$= (\sqrt{((g(x_0) - y_0)^2 + (g(x_1) - y_1)^2 + \ldots + (g(x_n) - y_n)^2})^2 =$$
$$= \sum_{i=1}^{n} (g(x_i) - y_i)^2 = \sum_{i=0}^{n} \left[(\sum_{j=0}^{k} a_j g_j(x_i)) - y_i \right]^2 \to \min$$

Ищем минимум функции:

$$\frac{\partial S}{\partial a} = 0$$
$$\frac{\partial S}{\partial a_j} = \sum_{i=0}^{n} 2 \left[\sum_{i=0}^{k} a_j \cdot g_j(x_i) - y_i \right] (0 + \ldots + 0 + g_l(x_i) + 0 + \ldots + 0) = 0$$
$$\sum_{i=0}^{n} \sum_{j=0}^{k} a_j \cdot g_j(x_i) g_l(x_i) = \sum_{i=0}^{n} y_i g_l(x_i)$$
$$\sum_{j=0}^{k} a_j \sum_{i=0}^{n} g_j(x_i) g_l(x_i) = \sum_{i=0}^{n} y_i g_l(x_i) ----- (6.1)$$

Теорема 6.1:

Так как СЛАУ (6.1) - квадратная ((k+1)*(k+1)), а матрица коэффициентов C_{pj}- не вырожденная, то система имеет всегда единственное решение (необходимо, чтобы функции g_j были линейно независимы в точках x_i)

$$\begin{cases} c_{00}a_0 + c_{01}a_1 + \ldots + c_{0k}a_k = d_0 \\ c_{10}a_0 + c_{11}a_1 + \ldots + c_{1k}a_k = d_1 \\ \ldots. \\ c_{k0}a_0 + c_{k1}a_1 + \ldots + c_{kk}a_k = d_k \end{cases} ----- (6.2)$$

$$c_{lj} = \sum_{i=0}^{n} g_j(x_i) g_l(x_i)$$
$$dl = \sum_{i=0}^{n} y_i g_l(x_i)$$

Теорема 6.2: решение СЛАУ (6.1) – всегда точка строгого глобального min функции S.

Пример: $C=\{a_0\}$ $g_0=1$

$C_{oo} \cdot a_0 = d_0$

$C_{00} = \sum_{i=1}^{n} 1*1 = n+1$

$d_0 = \sum_{i=0}^{n} y_i$

$a_0 = \frac{\sum_{i=0}^{n} y_i}{n+1}$ - среднее арифметическое

$C=\{a_0+a_1x\}$ $g_0=1$ $g_1=x=x_1$

$$\begin{cases} c_{00}a_0 + c_{01}a_1 = d_0 \\ c_{10}a_0 + c_{11}a_1 = d_1 \end{cases}$$

$C_{00}=\sum_{i=1}^{n} 1*1$

$C_{01}=\sum_{i=1}^{n} 1*x_i^1$

$C_{10}=\sum_{i=1}^{n} x_i*1$

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

$d_0 = \sum_{i=0}^{n} y_i*1$

$d_1 = \sum_{i=0}^{n} y_i*x_i$

П.2. Аппроксимация по ортонормированным наборам функций.

Предположим, что набор функций $\mathbf{g_0(x),g_1(x),...,g_k(x)}$ образует ортонормированную систему на сетке $\mathbf{x_0,x_1,...,x_n}$.

$\mathbf{(g_k,g_l)=g_k(x_0)g_l(x_0)+...+g_k(x_n)g_l(x_n)}=\sum_{i=0}^{n} g_k(x_i)g_l(x_i)$

(ортонормированная система – $\mathbf{(g_k,g_l)}$) = $\begin{cases} 0, k \neq l \\ 1, k = l \end{cases}$.

Следовательно, для ортонормированной системы функций матрица коэффициентов $\mathbf{C_{lj}}$ в системе (6.2) будет единичная и aj легко находится.

$\mathbf{a_j=d_j=(y_i,g_j)}$.

<u>Вывод</u>:

Для ортонормированной системы функций коэффициенты аппроксимации находятся гораздо проще, чем обычно. Основная проблема: получить ортонормированную систему функций?

1. использовать тригонометрическую интерполяцию - теорема (3.7).

Для ортонормированной системы функций:

$g(x)=\sum_{j=0}^{k}(y,g_j)g_j(x)$ (6.3)

g(x) – начальный интервал ряда Фурье функции **y**.

Если бы функция $\mathbf{g_j}$ образовывала не только ортонормированную систему, но и базис **(n+1)** мерного пространства, то **к=n** и это было бы в точности разложение Фурье.

Если система функций $\mathbf{g_j}$ не ортонормированная, то ее можно такой сделать с помощью процесса ортогонализации Грамма – Шмита.

П.3. Процесс ортогонализации Грамма – Шмита.

Рассмотрим не ортонормированную, линейно независимую систему векторов $\mathbf{h_0,h_1,\dots,h_k}$ преобразуем ее в ортонормированную за k шагов.
Рассмотрим **i-** ый шаг **(i=1,2,...,k)**:
Первый **i-** ый вектор уже ортонормирован:
$((\mathbf{l_0,l_1,\dots,l_{i-1}})$**-** ортонормирован, $(\mathbf{h_i,h_{i+1},\dots,h_k})$**-** произволен) – **i-** ый шаг.
Сделаем $\mathbf{h_i}$- перпендикулярно ко всем $\mathbf{l_0,l_1,\dots,l_{i-1}}$, а затем нормируем его (т.е. сделаем его длину равной единице).
1. "перпендикуляризация"
рассмотрим вектор:

$$e_i^* = h_i - \sum_{k=0}^{i-1}(h_i, l_k)e_k$$

Докажем, что $e_i^* \perp$ $\mathbf{l_0,l_1,\dots,l_{i-1}}$, имеем e_i^* скалярно домноженный на $\mathbf{e_j}$

$$\forall_j = 0,1,\dots,i-1 (e_i^*, e_j) = (h_i - \sum_{k=0}^{i-1}(h_i, e_k)e_k, e_j) =$$

$$= (h_i, e_j) - \sum_{k=0}^{i-1}(h_i, e_k)e_k, e_j) = (h_i, e_j) - \sum_{k=0}^{i-1}(h_i, e_k)[(e_k, e_j) = 0, кроме (e_j, e_j) = 1] = (h_i, e_j) - (h_i, e_j) = 0$$

2. "нормирование"

$e = \dfrac{e_i^*}{\|e_i^*\|_2}$ он ортогонален всем предыдущим.

Замечание:
$\|e_i^*\|_2 \neq 0$, так как $e_i^* \neq 0$ являясь линейной комбинацией линейно независимых векторов с нулевыми коэффициентами.

Тема 7: Нелинейная оптимизация.
Метод градиента (метод наискорейшего спуска).

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

Задача нелинейной оптимизации – найти [min f(x)]-одномерную ЗНО или [max f(x)]- многомерная ЗНО
x=(x1,x2,...,xn)
Как известно из высшей математики, экстремум функции f достигается (если функция дифференцируемая, гладкая) или на границе области D или в точках $\mathbf{x}_{\text{оптимальное}}$,в которых все частные производные обращаются в 0.

$$\begin{cases} \frac{\partial f}{\partial x_1} |_{x=x_{jgnbv}} = 0 \\ : \\ \frac{\partial f}{\partial x_n} |_{x=x_{jgnbv}} = 0 \end{cases} \quad (7.1),$$

Первая возможность рассматривать не будем.

$$\begin{cases} \frac{\partial f}{\partial x_1}(x_1 ... x_n) = 0 \\ : \\ \frac{\partial f}{\partial x_n}(x_1 ... x_n) = 0 \end{cases} \quad (7.2),$$

т.е. задачу нелинейной оптимизации (7.1) свели к задаче (7.2) – системе нелинейных уравнений.

Обратная процедура (СНУ $\rightarrow$ЗНО)

Рассмотрим СНУ:

$\begin{cases} f_1(x_1 ... x_n) = 0 \\ : \\ f_n(x_1 ... x_n) = 0 \end{cases}$ и функцию: $f(x_1 ... x_n) = \sum_{i=1}^{n} f_i^2 (x_1 ... x_n)$ (7.4), так как f всюду ≥ 0, то ее min достигается в точке $x=(x_1 \ldots x_n)$ являющейся, решением (7.3).

Сводить ЗНО к СНУ и решать ее м. Ньютона хорошо теоретически, но весьма проблемно практически, т.к. метод Ньютона может расходиться и функция f аналитически не задается, поэтому лучше использовать численное дифференцирование. При попытке численного дифференцирования возникает погрешность, из-за которой м.Ньютона теряет скорость и на каждом шаге приходится делать много вычислений. Поэтому на практике при решении ЗНО обычно используют специальные методы, самый распространенный – метод градиента.

<u>П.2. Метод градиента.</u>

Имеем min функции $f(x_1, \ldots, x_n)$ (без ограничений на область D или в предположении, что она достаточно большая, т.е. не помешает процессу).

<u>Идея метода градиента:</u>

1. Выбрать точку x^0 достаточно близкую к точке min;
2. Найти направление наибольшего убывания f в точке x (направление наискорейшего возрастания f в точке x^0 – вектор градиента (grad)

   $gradf |_{x=x_0} = \left(\frac{\partial f}{\partial x_1}\bigg|_{x=x_0}, ..., \frac{\partial f}{\partial x_n}\bigg|_{x=x_0} \right)$, направление: поиск убывания – противоположный градиенту: вектор – grad f $|_{x=x^0}$.
3. Двигаемся по направлению градиента (по прямой) до тех пор, пока не начнем подниматься, (т.е. пока не достигнем min на нашей одномерной траектории в точке x^1).

4. Из точки x^1 двигаемся в направлении наискорейшего убывания ($-grad\, f\,|_{x=x^0}$) до точки x^2. Продолжаем этот процесс, до тех пор, пока не достигнем заданного min с заданной точностью. Критерий прерывания универсальный.

Переведем алгоритм на математический язык:

$g_0(t) = f(x_0 - t\,gradf\,|_{x=x^0})$ - траектория (зависит от t) движения на первом этапе.

Ищем min $g_0(t)$:

а) Предположим, что min достигается при $t=t_0$:

$x^1=x^0 - t\ grad\ f\,|_{x=x^0}$

б) Рассмотрим $g1(t)= f(x^1 - t\ grad\ f\,|_{x=x^1})$

$x^2=x^1 - t_1\ grad\ f\,|_{x=x^1}$,

и т.д.

Т.е. на каждом этапе метода градиента необходимо решать ЗНО (находить min $g_i(t)$).

Для нахождения градиента, функцию f необходимо дифференцировать (аналитически или численно).

П.3. Одномерная ЗНО. Метод деления.

Первый вариант решения одномерной ЗНО - свести ее к нелинейному уравнению:

$\frac{dg_i}{dt} = 0$ и решать его любым известным методом.

Этот метод применим, если g_i гладкая и ее производную нетрудно явно вычислить. Часто это не удается и поэтому одномерную ЗНО решают именно как ЗНО (т.е. в том виде, как она дана).

Будем решать одномерную ЗНО (ищем min g(t)) следующим образом:

Предположим, нам известно, что min значение $t_{\text{оптимальное}} \in [a,b]$. Разобьем интервал [a,b] точками: t_i ($a=t_0<t1<\ldots<t_n=b$) на маленькие интервалы и вычислим $g(t_i)$, выберем среди них min – $g(t_{i\ \text{опт.}})$.

Тогда, если g имела один min на [a,b], то: $t_{i\ \text{опт.}-1} \le t_{i\ \text{опт.}} \le t_{i\ \text{опт.}+1}$.

На следующем шаге берем в качестве $a=t_{i\ \text{опт.}-1}$, в качестве $b=t_{i\ \text{опт.}+1}$, разбиваем полученный интервал на части и производим очередное уточнение для $t_{\text{оптимального}}$. Процесс продолжается до тех пор, пока не будет достигнута заданная точность.

Число интервалов должно быть ≥3, чтобы интервал поиска уменьшался.

При подобном методе деления на каждом шаге необходимо искать значение функции f в двух точках и длина интервала поиска уменьшается приблизительно вдвое. Для достижения заданной точности ε нам потребуется проделать $\left[\log_2 \frac{b-a}{\varepsilon}\right]$ шагов и $N = \left[\log_2 \frac{b-a}{\varepsilon}\right] * 2$ операций (7.3).

П.4. Метод золотого сечения.

Метод золотого сечения применяется для минимизации числа действий при решении ЗНО методом деления. При этом $[\overset{a\ \ x}{\cap \|} \overset{x\ \ b}{\cap}]$ разбивается на три интервала и выбирается $\min g(t_i)$. На следующем шаге один из крайних отрезков будет обращен и остается $\overset{a_1}{|}\ \overset{!}{|}\ \overset{?}{|}\ \ \overset{b_1}{|}$.

При удачном подборе пропорции деления первоначального интервала [a,b] и делении отрезка [a1,b1] в тех же пропорциях, потребуется вычислить значения функции f только в одной точке, потому что значения в точке нам было известно на предыдущем шаге.

<u>Подберем необходимые пропорции:</u>

Считаем, что длина [a,b]=1 и от этого интервала были отсечены два одинаковых отрезка длины x. Тогда, если отрезок обозначим за y, то получим:

$\frac{y}{x}=\frac{1-x}{1}$, y=(1-x)x, но с другой стороны y=1-2x

$$1-2x = x(1-x)$$

$$x=\frac{3\pm\sqrt{5}}{2}$$

$x=\frac{3\pm\sqrt{5}}{2}\approx 0{,}39$, при этом интервал [a,b] уменьшается в $\left(\frac{1}{1-x}\right)$ раз, что составляет:

$\frac{1}{1-x}=\frac{\sqrt{5}+1}{2}\approx 1{,}61$.

<u>Трудоемкость одномерного ЗНО методом золотого сечения.</u>

Нам потребуется $N=\left[\log_2\frac{b-a}{\varepsilon}\right]*1$ шагов (7.4). Данное N меньше числа N из формулы (7.3). Выигрыш происходит за счет того, что мы на каждом шаге значение функции считаем 1 раз и, несмотря на то, что длина интервала уменьшается не в 2 раза, а только лишь в 1,618 раза, происходит выигрыш по количеству итераций.

<u>Тема 8. Применение метода Монте - Карло в численных методах.</u>

<u>П.0. Постановка и пример задачи, решаемой методом Монте-Карло.</u>

Методом М.-К. можно решать самые разные задачи численных методов (вычислить интеграл, решить линейные и не линейные системы уравнений и дифференциальные уравнения).

Будем вычислять этим методом интегралы.

Обычные методы численного интегрирования (см. тема 4 §1) хорошо работают только для интегралов небольшой кратности, выше рассматривали интегралы только первой кратности, например, для вычисления десятичного интеграла [0,1]*…*[0,10] с шагом h=0,1 используя аналог квадратурных формул Ньютона – Котеса придется суммировать $(11)^{10}$ слагаемых, что практически нереально. Поэтому применяют метод М.-К..

<u>Характерная особенность метода М.-К.:</u>

Невозможность получения гарантированной оценки точности. Можно утверждать, что некоторая точность достигнута с определенной вероятностью.

Практическое правило:
Будем считать, что если вероятность $\geq 0{,}9$ (в зависимости от конкретной ситуации), то оценка является гарантированной.

П.1. Первый способ нахождения интервала методом М.-К..

Задача:
Найти n-кратный интеграл по n-мерному единичному кубу(по области D):
$\int \ldots \int f(x_1,\ldots,x_n)dx_1\ldots dx_n$ (8.1)
Предположим, что область D заключена в n- мерный параллелепипед:
$П=[a_1,b_1]*[a_2,b_2]*\ldots*[a_n,b_n]$
Будем "бросать" равномерно по всему параллелепипеду точки:
$X^{(i)}=(x_1^{(i)},\ldots,x_n^{(i)})$
Для этого первая координата x_1 должна быть равномерно распределена на интервале $[a_1,b_1],\ldots$, а x_n должна быть равномерно распределена на интервале $[a_n,b_n]$, и быть независимыми.
Обычно для этих целей используют стандартный датчик случайных чисел:
$random \to (b-a)random + a$

$[0,1] \to [a,b]$

Датчики случайных чисел бывают:
1. физические,
2. математические.

Часто используют математический ДСЧ :
Берется некоторое случайное число $\sqrt[q]{2} = 1{,}41421956147$

Если полученая последовательность шестизначных чисел будет равномерно распределена на интервале: $[0,..999999] \to [0,a_1,a_2,a_3,a_4,a_5,a_6]$, то интервал перейдет в интервал [0,1]. У подобного датчика случайных чисел имеется недостаток: он не совсем случаен. Обычно для избавления от этого на математический датчик накладывают физический датчик.
Интеграл (8.1) заменяем на многомерный объем параллелепипеда П:
$(8.1) \approx \dfrac{V(П)\sum\limits_{x^{(i)}\in D} f(x^{(i)})}{N}$,где: N – общее число испытаний, $x(i)\in D$ – успешные испытания.

$$I_0 = \int \ldots \int f(x_1,\ldots,x_n)dx_1\ldots dx_n \approx \frac{V(П)\sum\limits_{i=1}^{N} f(x^{(i)}\chi D(x^{(i)}))}{N} \qquad (8.2)$$

χ - функция, характеризующая область D.

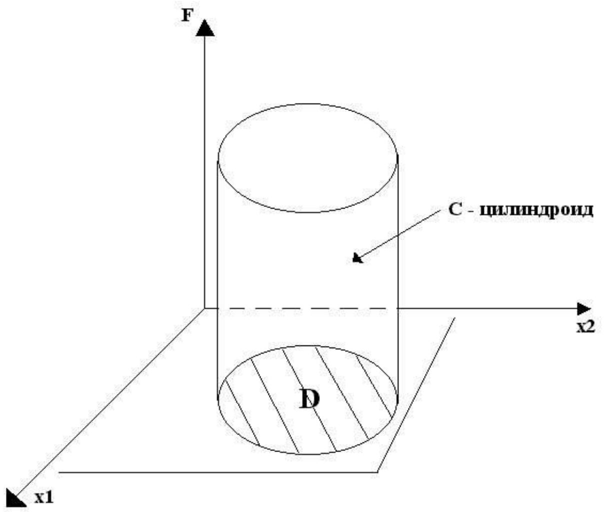
$$\chi D(x) = \begin{cases} 1, x \in D \\ 0, x \notin D \end{cases}$$

На основании теорем из теории вероятностей можно доказать, что:
$P\left(\left|I_0 - I_{np}\right| < \dfrac{\varepsilon}{\sqrt{N}}\right) > 1 - const(f,D,\varepsilon)$ (8.3).
Практический вывод из (8.3):

Для увеличения точности в k раз необходимо увеличить количество испытаний в k^2 раз.
Const – оценивается весьма сложно. Точные оценки для вероятности приведем для второго способа вычисления интеграла.

П.2. Второй способ вычисления интеграла методом М.-К..

Изначально считаем, что :$D \subset [0,1]^n$ и всюду в области D;
$0 \le f(x) \le 1$- в этом случае интеграл (8.1) можно заменить, на объем (n+1) мерной фигуры лежащей под графиком функции:



$I_0=V(c)$
Для нахождения I_0 будем "бросать" точки $Z^{(i)}=(x^{(i)},y^{(i)})$, равномерно распределенные в $[0,1]^n*[0,1]$. При этом $V(c) \approx \frac{n}{N}$, где: n – число попаданий точки $Z^{(i)}$ в область C, N – общее число испытаний.
$Z^{(i)} \in C \Leftrightarrow x^{(i)} \in D, 0 \le y^{(i)} < f(x^{(i)})$.
На основании неравенства Чебышева приходим к выводу, что:
$P\left(\left|I_0 - \frac{n}{N}\right| \le \varepsilon\right) \ge 1 - \frac{D(\xi)}{\varepsilon^2} = 1 - \frac{I_0(1-I_0)}{\varepsilon^2 N} \ge 1 - \frac{1}{4\varepsilon^2 N}, \quad \xi = \frac{n}{N}, \quad D\xi = \frac{I_0(1-I_0)}{N}$. (6 балов).