Химический факультет
Кубанского государственного университета

Численные методы и программирование

Часть II. Численные методы решения уравнений и систем нелинейных уравнений. Обработка табличных данных

Методическое пособие по курсу
«Численные методы и программирование»
(для студентов химического факультета)

Краснодар 2002

Печатается по решению учебно-методической комиссии химического факультета

Представлено кафедрой общей и неорганической химии

Составители: И.В. Сухно
В.А. Волынкин
В.Ю. Бузько

Редактор: В.Т. Панюшкин, профессор, д.х.н.

ЧИСЛЕННЫЕ МЕТОДЫ РЕШЕНИЯ НЕЛИНЕЙНЫХ УРАВНЕНИЙ

Корни линейных и квадратных уравнений можно найти по известным формулам. Хотя алгебраические уравнения третьей и четвертой степени еще можно решить аналитическими методами, соответствующие формулы достаточно сложны. Например, нельзя аналитически (точно) найти корни уравнений

$x^6+4x^5-5x^4+x^3+3x^2-9x+11=0$ (алгебраическое уравнение)
$x^2-\sin x=0$ (трансцендентное уравнение),

но с помощью численных методов решения можно найти приближенные корни уравнений.

ЗАМЕЧАНИЕ

Корнем уравнения $f(x)=0$ называется такое значение аргумента, при котором $f(x)$ обращается в нуль.

Метод деления отрезка пополам (метод полуинтервалов, метод дихотомии)

Пусть нам удалось найти отрезок [a,b], в котором расположено искомое значение корня $x=c$, т.е. $a<c<b$. В качестве начального приближения корня c_o примем середину этого отрезка: $c_o=(a+b)/2$. Далее исследуем значения функции $f(x)$ на концах отрезков $[a,c_o]$ и $[c_o, b]$ в точках a, c_o, b. Тот из них, на концах которого $f(x)$ принимает значения разных знаков, содержит корень,

поэтому его принимаем в качестве нового отрезка. В качестве следующего приближенного значения корня принимаем середину нового отрезка и т.д. Каждый новый шаг называется итерацией, а сама методика последовательного приближения к корню называется итерационным методом. После каждой итерации отрезок, на котором расположен корень, уменьшается вдвое, то есть после n итераций он сокращается в 2^n раз.

Пусть $f(a)<0$, $f(b)>0$. Пусть начальное приближение $c_o=(a+b)/2$, т.к. $f(c_o)<0$, то $c_o<c<b$ и рассматриваем отрезок $[c_o,b]$. Следующее приближение $c_1=(c_o+b)/2$, т.к. $f(c_1)>0$, то $c_o<c<c_1$ и рассматриваем отрезок $[c_o,c_1]$. Следующее приближение $c_2=(c_o+c_1)/2$ и т.д. (рис.1).

Итерационный процесс продолжается до тех пор, пока значение функции f(x) после n-й итерации не станет меньшим по модулю некоторого заданного малого ξ, т.е. $|f(c_n)| < \xi$. Можно так же потребовать, чтобы расчет прекратился, когда точность вычисления корня составит 1/1000000 от исходного отрезка; или задать число итераций.

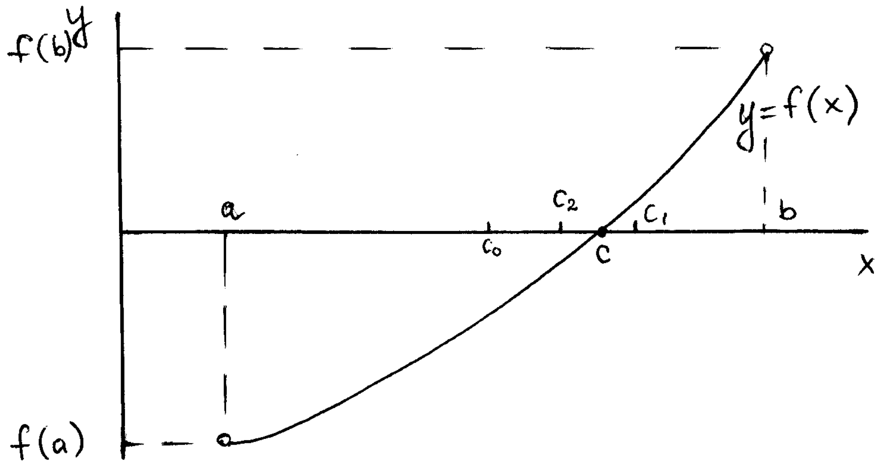


Рис.1

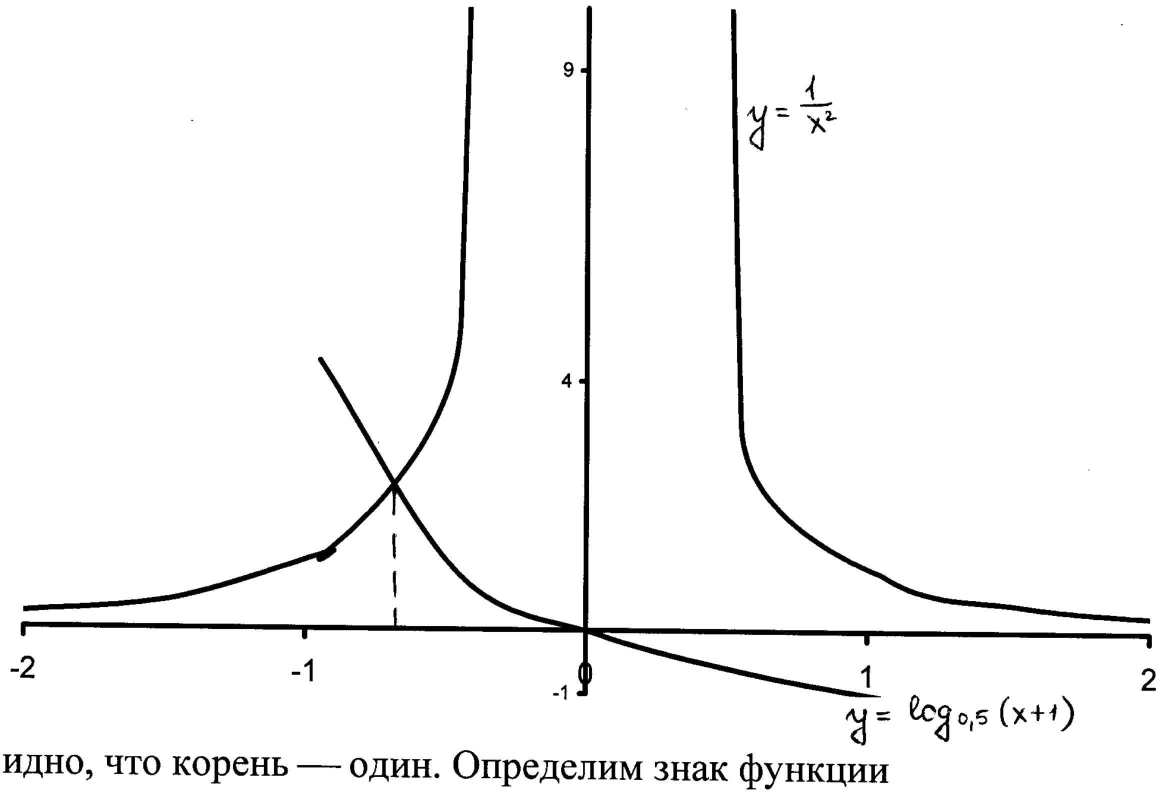
Метод деления отрезка пополам довольно медленный, однако, этот итерационный процесс всегда сходится, т.е. при его использовании решение получается всегда, т.к. при каждой последующей итерации происходит приближение к корню.

Пример:

Найти с точностью ξ=0,01 корень уравнения $x^2 \bullet \log_{0,5}(x+1)=1$.

Решение:

Сначала определим отрезок существования корня — отделим корень графически: $\log_{0,5}(x+1)=\frac{1}{x^2}$.



Видно, что корень — один. Определим знак функции

$$f(x)=x^2+\log_{0.5}(x+1)-1=x^2\frac{\lg(x+1)}{\lg 0.5}-1=x^2\frac{\lg(x+1)}{-0.31}-1$$

слева и справа от корня

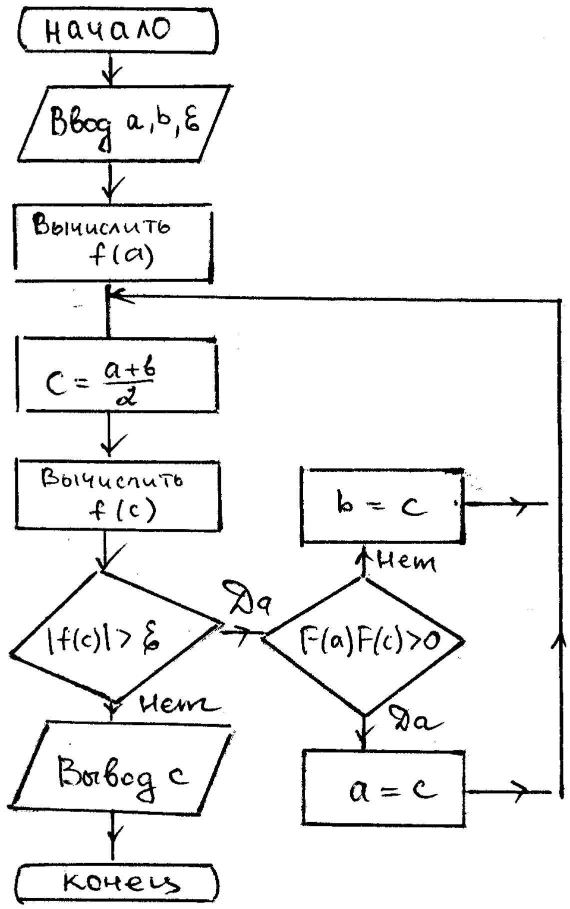
x	-0.8	-0.5
f(x)	+	-

Далее уточним корень с точностью $\xi=10^{-2}$:

n	a_n	b_n	$x_n=(a_n+b_n)/2$	$f(a_n)$	$f(b_n)$	$f(x_n)$
0	-0.8	-0.5	-0.65	0.486182	-0.74998	-0.36
1	-0.8	-0.65	-0.725	0.486182	-0.36003	-0.0209
2	-0.8	-0.725	-0.7625	0.486182	-0.02093	0.20596
3	-0.7625	-0.725	-0.74375	0.205957	-0.02093	0.08673
4	-0.74375	-0.725	-0.734375	0.086731	-0.02093	0.03155
5	-0.734375	-0.725	-0.7296875	0.031547	-0.02093	0.00498
6	-0.7296875	-0.725	-0.72734375	0.004981	-0.02093	-0.0081

Корень ≈ -0.73

Блок-схема метода



Метод хорд (метод ложного положения, метод линейного интерполирования, метод пропорциональных частей)

В этом методе нелинейная функция f(x) на определенном интервале [a,b] заменяется линейной, в качестве которой берется хорда - прямая, стягивающая концы нелинейной функции. Хорда определяется как прямая, проходящая через точки с координатами (a,f(a)) и (b,f(b)). Имея уравнение хорды y=cx+d, можно найти точку ее пересечения с горизонтальной осью (y=0 и найдя x). Полученную точку x_1 считают новой границей отрезка, где содержится корень и т.д. Получают последовательность приближений

$x_2, x_3, \ldots$ сходящуюся к корню (рис.2). Метод применим только для монотонных функций.

Алгоритм метода зависит от свойств функции f(x). Если $f(b) \cdot f''(b) > 0$, то строящаяся на каждом этапе хорда имеет правый фиксированный ("закрепленный") конец, и алгоритм имеет вид:

$$x_{i+1} = x_i - \frac{f(x_i)}{f(b) - f(x_i)} \bullet (b - x_i)$$

При этом последовательность $x_1, x_2, \ldots$ приближается к корню слева.

Если $f(a) \cdot f''(a) > 0$, то строящаяся на каждом этапе хорда имеет левый фиксированный ("закрепленный") конец, и алгоритм имеет вид:

$$x_{i+1} = a + \frac{f(a)}{f(a) - f(x_i)} \bullet (x_i - a)$$

При этом последовательность $x_1, x_2, \ldots$ будет приближаться к корню справа.

Теоретически доказано, что если первые производные на концах интервала при монотонной и выпуклой функции f(x) не различаются более чем в 2 раза, то справедливо соотношение $|x^* - x_i| < |x_i - x_{i-1}|$ и условием прекращения пополнения последовательности может быть $|x_{i+1} - x_i| \le \xi$, а в качестве корня принято x_{i+1}. Можно также окончить процесс и при достижении $f(x_i) \le \xi$. На практике можно указанные условия применять и без предварительной проверки производных.

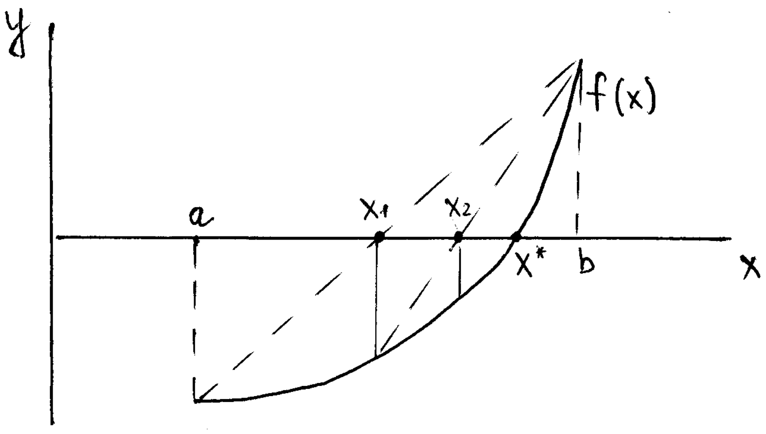
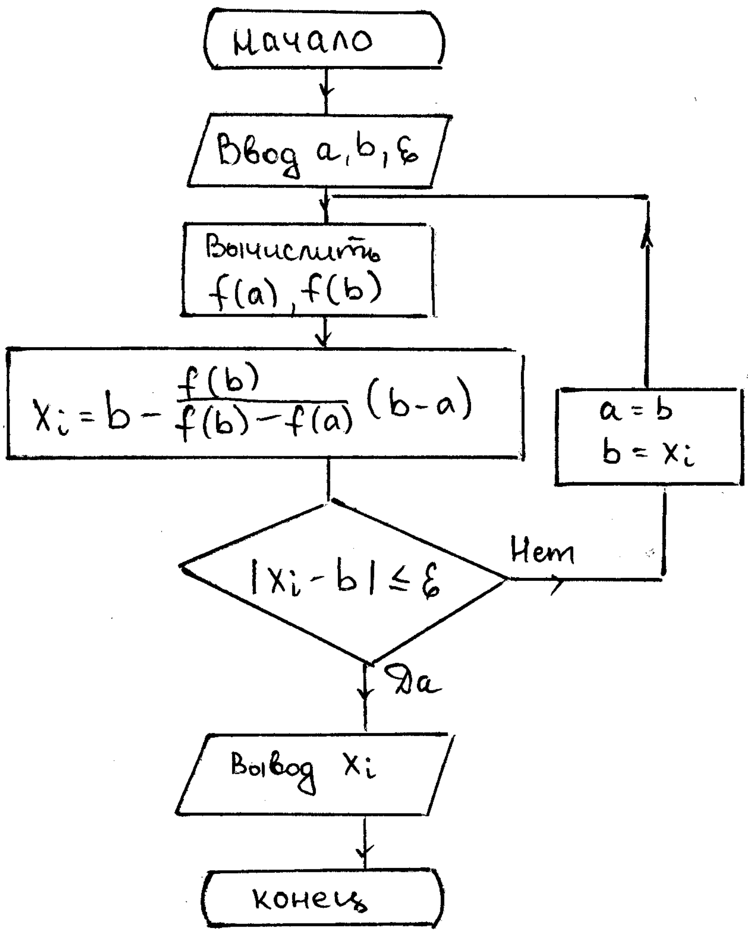


Рис.2

Блок-схема метода



Пример:

Найти с точностью $\xi=10^{-3}$ меньший корень уравнения $x^3+3x^2-3=0$

Решение:

Определим отрезок существования меньшего корня – отделим корень. Отделим корни аналитически: приравняем $f'(x)=0$

$(f(x)=x^3+3x^2-3=0)$

$f'(x)=3x^2+6x=0$

$x_1=0 \quad x_2=-2$

Составим таблицу знаков функции f(x) , полагая x равным

- критическим значением (корни производной) или ближайшим к ним;
- граничным значениям (исходя из области допустимых значений аргумента).

x	$-\infty$	**-2**	**-1**	**0**	**1**	$+\infty$
f(x)	-	+	-	-	+	+

Левый корень лежит в интервале $(-\infty;-2)$, для уточнения интервала найдем f(-3)=-3, следовательно

x	-3	-2	-1	0	1
f(x)	-	+	-	-	+

Т.о. корни содержатся в интервалах: (-3,-2); (-2,-1); (0,1).

Уточним корень методом хорд в интервале (-3,-2), т.е. a=-3, b=-2.

Найдем вторую производную $f''(x)=6x+6$, $f''(x)<0$ на интервале (-3;-2), поэтому используем формулу

Пример:

Найти с точностью $\xi=10^{-3}$ меньший корень уравнения $x^3+3x^2-3=0$

Решение:

Определим отрезок существования меньшего корня – отделим корень. Отделим корни аналитически: приравняем $f'(x)=0$

$(f(x)=x^3+3x^2-3=0)$

$f'(x)=3x^2+6x=0$

$x_1=0 \quad x_2=-2$

Составим таблицу знаков функции f(x) , полагая x равным

- критическим значением (корни производной) или ближайшим к ним;
- граничным значениям (исходя из области допустимых значений аргумента).

x	-∞	**-2**	**-1**	**0**	**1**	+∞
f(x)	-	+	-	-	+	+

Левый корень лежит в интервале (-∞;-2), для уточнения интервала найдем f(-3)=-3, следовательно

x	-3	-2	-1	0	1
f(x)	-	+	-	-	+

Т.о. корни содержатся в интервалах: (-3,-2); (-2,-1); (0,1).

Уточним корень методом хорд в интервале (-3,-2), т.е. a=-3, b=-2.

Найдем вторую производную $f''(x)=6x+6$, $f''(x)<0$ на интервале (-3;-2), поэтому используем формулу

$$x_{i+1} = a + \frac{f(a)}{f(a) - f(x_i)} \bullet (x_i - a)$$, где x_o=b=-2, f(a)=f(-3)=-3

i	x_i	$f(x_i)$	x_i-a
0	-2	1	1
1	-2.25	0.796875	0.75
2	-2.4074074	0.434435808	0,592593
3	-2.4823669	0.189730569	0.517633
4	-2.5131566	0.074881701	0.486843
5	-2.5250125	0.028372102	0.474987
6	-2.5294626	0.010583631	0.470537
7	-2.5311176	0.003925032	0.468883
8	-2.5317294	0.001452481	0.468271
9	-2.531956		

Корень ≈-2,532.

Метод Ньютона (метод касательных)

В этом методе нелинейная функция f(x) на отделенном интервале [a,b] заменяется линейной, в качестве которой берется касательная, проводимая в текущей точке последовательности. Уравнение касательной находится по координате одной точки и углу наклона (значение производной). В качестве начальной точки в зависимости от свойств функции берется или левая точка x_o=a, если $f(a)\cdot f''(a)>0$, или правая точка x_o=b, если $f(b)\cdot f''(b)>0$.

Алгоритм записывается следующим образом:

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

Получаем последовательность приближенных значений $x_1, x_2, \ldots$, каждый член которой ближе к истинному корню x^*, чем предыдущий (рис.3).

Если производная $f'(x)$ мало изменяется на отрезке [a,b], то используют упрощенный вариант:

$$x_{i+1} = x_i - \frac{f(x_i)}{f'(x_0)}$$

(геометрически это означает, что касательные в точках $(x_i, f(x_i))$ заменяются прямыми, параллельными касательной, проведенной к кривой $y=f(x)$ в точке $(x_o, f(x_o))$).

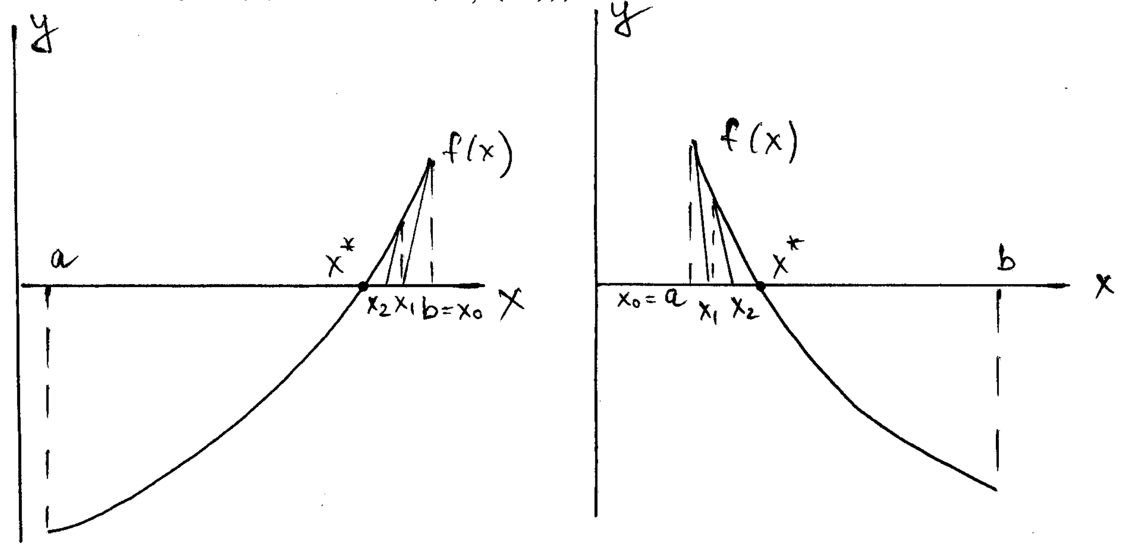


Рис.3

Условия окончания поиска аналогичны методу хорд. Главным достоинством метода является то, что на каждой итерации погрешность возводится в квадрат, т.е. число верных знаков корня удваивается. Если начальное приближение выбрано удачно, то после 5-6 итераций погрешность станет величиной $\sim 2^{-64}$. Для получения такой малой погрешности в методе дихотомии необходимо ~50 итераций.

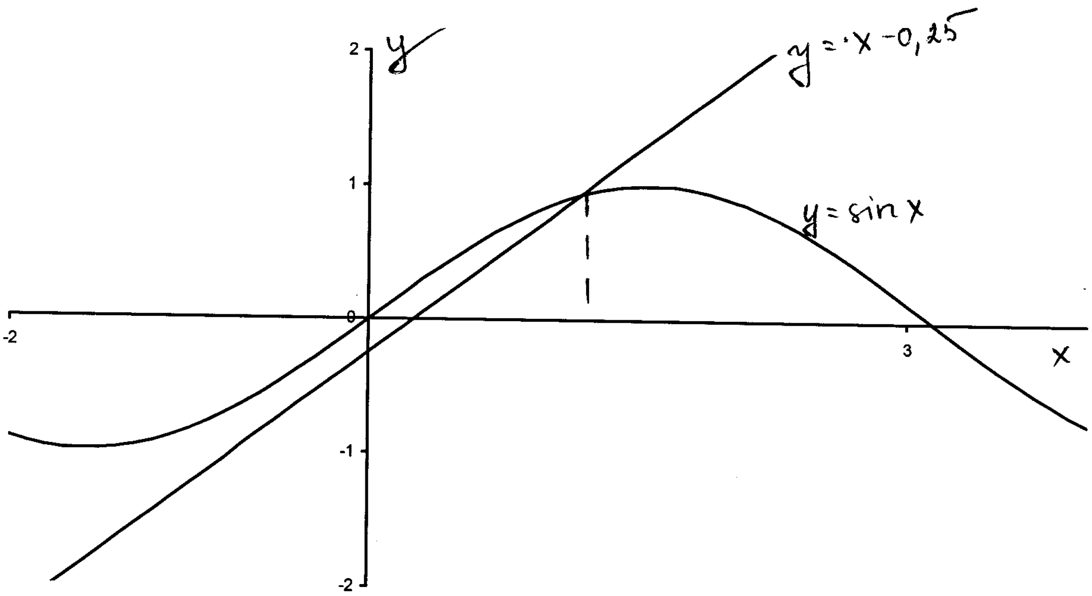
Пример:

Методом Ньютона уточнить до ξ=0,0001 корень уравнения

$x - \sin(x) = 0{,}25$.

Решение:

Отделим корень графически.



Корень находится в интервале [0,982; 1,178]: a=0,982, b=1,178.

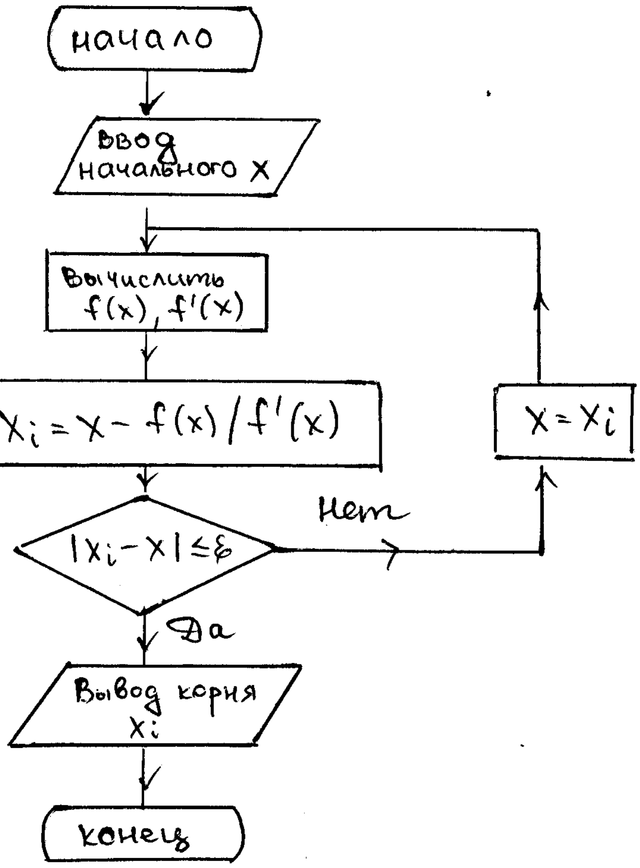
$f'(x) = 1 - \cos(x)$

$f''(x) = \sin(x) > 0$ на [0,982; 1,178], $f(1{,}178) \cdot f''(x) > 0$, поэтому $x_0 = 1{,}178$.

i	x_i	$f(x_i)=x_i-\sin(x_i) - 0.25$	$f'(x_i)=1 - \cos(x_i)$
0	1.178	0.00416	0.61723
1	1.1715	0.00017	0.61123
2	1.1713	0.00003	0.61110
3	1.17125		

Корень ≈1,1712

Блок-схема метода



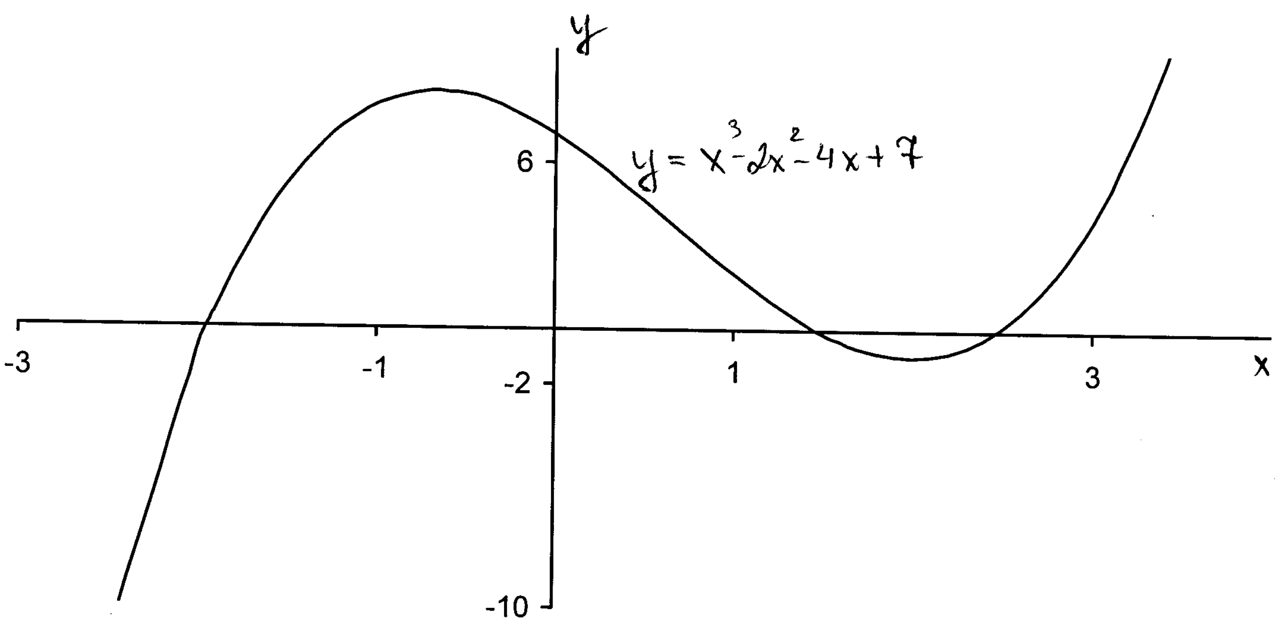
Комбинированный метод

Метод также базируется на замене нелинейной функции f(x) линейной, но с учетом стремления к корню метода хорд и Ньютона с разных сторон. Для повышения эффективности используют оба алгоритма одновременно. Один шаг делается методом хорд, а следующий с другой стороны – методом Ньютона. Интервал, где содержится корень, сокращается с обеих сторон. Поиск можно прекратить, как только разница станет меньше предварительно заданной погрешности ξ.

Пример:

Уточнить корни уравнения $x^3-2x^2-4x+7=0$ с погрешностью $\xi<0{,}001$.

Решение:



Отделим корни графически:

Три корня лежат в интервалах $x_1\in[-2;-1]$, $x_2\in[1;2]$, $x_3\in[2;3]$.

Уточним корень x_1.

Учитывая, что $f(-2)<0$, $f(-1)<0$, $f''(x)=6x-4$ и при $-2\le x\le -1$, $f''(x)<0$, для расчетов примем следующие формулы:

$$x_{\Lambda(i+1)} = x_{\Lambda i} - \frac{f(x_{\Lambda i})}{f'(x_{\Lambda i})}$$

$$x_{\Pi(i+1)} = x_{\Lambda i} - \frac{f(x_{\Lambda i})}{f(x_{\Pi i}) - f(x_{\Lambda i})} \bullet (x_{\Pi i} - x_{\Lambda i})$$

Где $x_{\Lambda i}$ и $x_{\Pi i}$- соответственно значение корня по недостатку (слева) и избытку (справа), $x_{\Lambda o}$=-2 , $x_{\Pi o}$=-1.

i	$x_{\Lambda i}$	$x_{\Pi i}$	$x_{\Pi i}$-$x_{\Lambda i}$	$f(x_{\Lambda i})$	$f(x_{\Pi i})$
1	-2	-1	1	-1	8
2	-1.9400	-1.8900	0.0500	-0.0686	0.0645
3	-1.9355	-1.9353	0.0002	-0.0011	0.0020

x_1≈-1,935

За приближенное значение корня можно взять $\frac{1}{2}\cdot(x_{\Lambda i}+x_{\Pi i})$.

Метод параболической аппроксимации

В этом методе функция f(x)заменяется не линейной, а параболической функцией, что является более точной заменой. Следовательно, метод может обеспечить более быструю сходимость к решению. На первом этапе параболу обычно строят по трем точкам: крайним и средним точкам интервала (a,b), где отделен корень, т.е. (a,f(a)), ((a+b)/2, f((a+b)/2)), (b,f(b)).

По полученному уравнению параболы $y=c_2x^2+c_1x+c_o$ находят приближенный корень, для чего решают уравнение $c_2x^2+c_1x+c_o=0$. На втором этапе строят параболу по трем точкам: найденному приближенному корню и двум предыдущим точкам (слева и справа от этой точки), лежащим по разные стороны оси

x. Такой вариант выбора точек на практике быстрее приводит к решению по сравнению с вариантом, когда для построения параболы берутся последовательно три последние точки. Такая процедура повторяется до тех пор, пока величина отрезка, внутри которого находится корень, не будет меньше ξ– предварительно заданной погрешности.

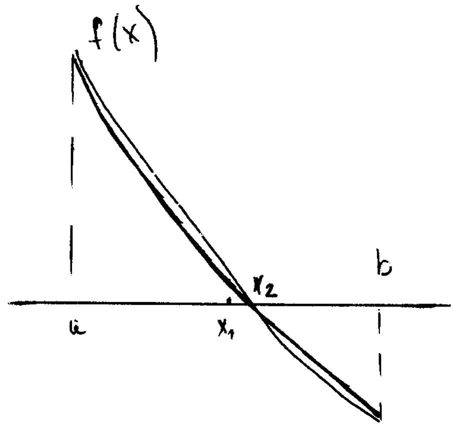


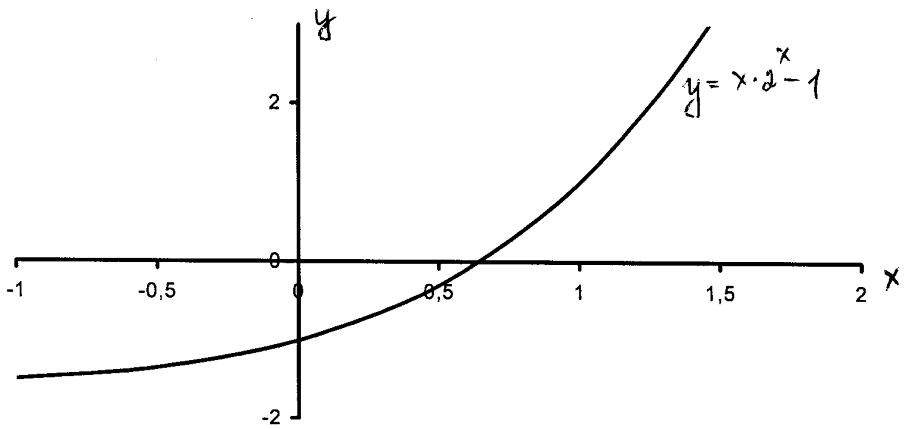
Рис.4

Пример одного шага уточнения корня методом параболической аппроксимации. Начальная парабола проведена через точки a,x_1,b (x_1-середина [a,b]); x_2-пересечение параболы с осью. Следующая парабола проводится через x_1,x_2,b.

Пример:

Для уравнения $x \bullet 2^x - 1 = 0$ сделать две итерации методом параболической аппроксимации.

Отделим корень графически.



Корень находится в промежутке [0,1], т.е. a=0, b=1.

Выбираем среднюю точку интервала x=1/2 и вычисляем значение функции в этих точках

x=0; f(0)=-1

x=0,5; f(0,5)≈-0,293

x=1; f(1)=1

Через эти три точки проводим параболу $y=c_2x^2+c_1x+c_o$. Для того, чтобы парабола проходила через заданные точки, коэффициенты c_i должны удовлетворять следующим уравнениям:

$$\begin{cases} 0^2 \bullet c_2 + 0 \bullet c_1 + c_0 = -1 \\ (1/2)^2 \bullet c_2 + 1/2 c_1 + c_0 = -0.293 \\ 1^2 \bullet c_2 + 1 \bullet c_1 + c_0 = 1 \end{cases} \qquad \begin{cases} c_0 = -1 \\ c_1 = -0.828 \\ c_2 = 1.172 \end{cases}$$

Для нахождения приближенного значения корня решим уравнение

$1,172x^2+0,828x-1=0$, что на участке (0,1) дает x=0.636.

Найдем значения функции в этой точке

$f(0,636)=0,636 \cdot 2^{0,636}-1=-0,011647$

Полученное значение не равно нулю, т.к. парабола приблизительно описывает исходную функцию.

Используя три точки: (0,5;-0,293); (0,636;-0,011647); (1;1) построим новую параболу. Коэффициенты этой параболы

$$\begin{cases} c_0 = -1 \\ c_1 = 0.828 \\ c_2 = 1.172 \end{cases}$$

Метод итераций (метод последовательных приближений)

Предварительно исходное уравнение f(x)=0 преобразуем к виду φ(x)=x. Затем выбирают начальное значение x_o и подставляют его в левую часть уравнения, но $\varphi(x_o)\neq x_o$, т.к. x_o взято произвольно и не является корнем уравнения. Полученное $\varphi(x_o)=x_1$ рассматривают как очередное приближение к корню. Его снова подставляют в левую часть уравнения $\varphi(x_1)$ и получают следующее значение x_2 ($x_2=\varphi(x_1)$) и т.д. В общем случае $x_{i+1}=\varphi(x_i)$. Получающаяся т.о. последовательность $x_o,x_1,x_2,\ldots$при определенных условиях может сходиться к корню x^* (рис.5).

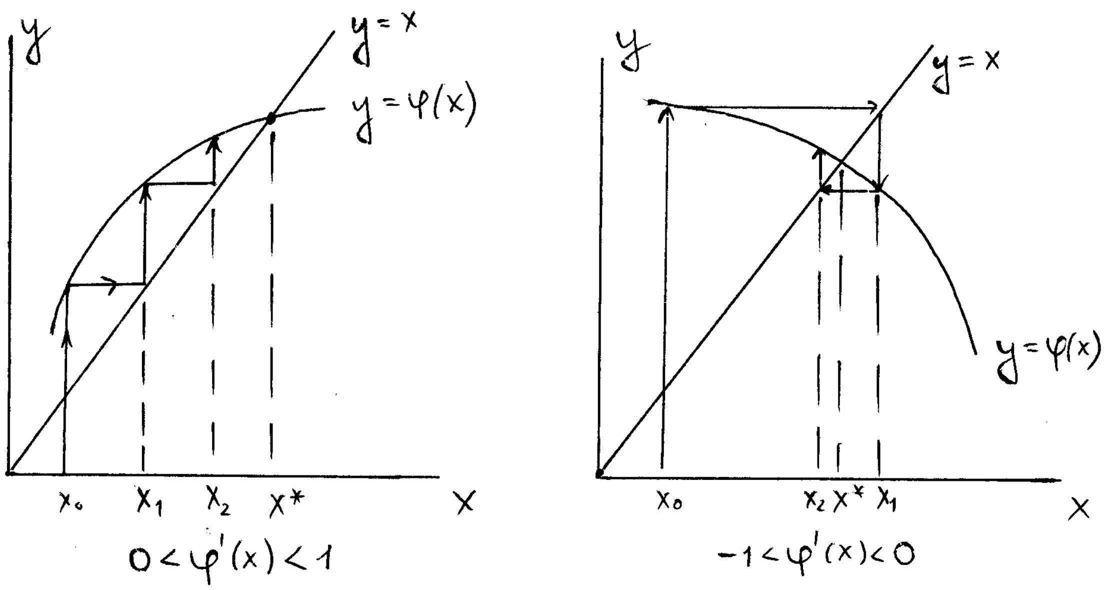


Рис.5

Таким условием является $|\varphi'(x)| \le 1$ на [a,b], причем, чем ближе модуль к нулю, тем выше окажется скорость сходимости к решению. В противном случае последовательность расходится от искомого решения (говорят, что "метод не сходится") (рис.6).

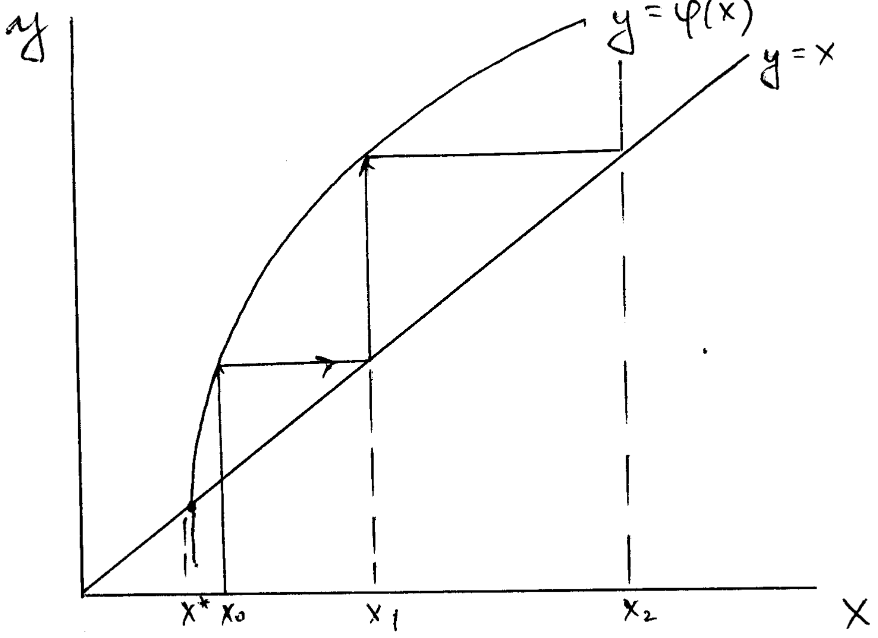


Рис.6

Видно, что последовательность $x_o,x_1,x_2,\ldots$ удаляется от корня. Это всегда будет иметь место, когда тангенс угла наклона $\varphi(x)$ в окрестности корня по модулю больше единицы.

Существуют различные способы преобразования уравнения $f(x)=0$ к виду $\varphi(x)=x$. Одни могут привести к выполнению условия сходимости всегда, другие - в отдельных случаях.

Самый простой способ:

$f(x)+x=0+x$, $f(x)+x=\varphi(x)$, $\varphi(x)=x$,

но он не всегда успешен.

Другой способ:

$\varphi(x)=x-f(x)/k$,

причем k выбирается так, чтобы $|k|>Q/2$, где $Q=\max\limits_{[a,b]}|f'(x)|$ и знак k совпадал бы со знаком $f'(x)$ на [a,b]. Погрешность можно оценить из соотношения:

$$|x^*-x_i|\le\frac{q}{1-q}|x_i-x_{i+1}|, \text{где } q=\max_{[a,b]}\varphi'(x).$$

Вследствие этого для окончания вычислений в методе итераций применяют соотношение $\frac{q}{1-q}|x-x_{i+1}|\le\xi$.

Часто используют упрощенное окончание поиска $|x_i-x_{i+1}|\le\xi$, не вычисляя максимальное значение производной, но в этом случае погрешность решения может не соответствовать заданной (т.е. быть больше или меньше).

Пример:

Уточнить корень погрешностью $\xi<0{,}001$ уравнения $2x+\lg(2x+3)=1$

Решение:

Корень уравнения $f(x)=2x+\lg(2x+3)-1=0$ находится в [0;0,5], т.е. a=0, b=0,5.

Приведем уравнение к виду, удобному для итераций $\varphi(x)=x$. Функцию $\varphi(x)$ будем искать из соотношения $\varphi(x)=x-f(x)/k$, считая для повышения сходимости, что $|k|\ge Q/2$, где $Q=\max\limits_{[0;0.5]}|f'(x)|$; число k имеет тот же знак, что и $f'(x)$ на [0;0,5].

Находим:

$$f'(x)=2+\frac{0.8686}{2x+3},\quad Q=\max_{[0;0.5]} f'(x)=2+\frac{0.8686}{2\bullet 0+3}\approx 2.2895$$

$f'(x)>0$ при $0\le x\le 0{,}5$.

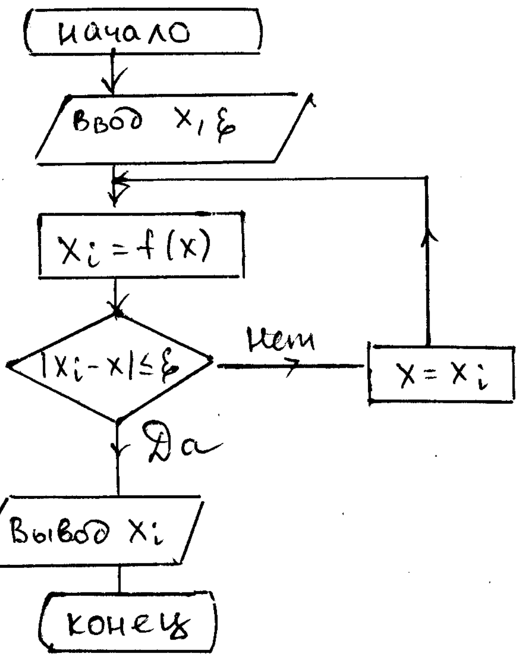
Примем k=2, тогда $\varphi(x)=x-f(x)/2=0{,}5-0{,}5\cdot\lg(2x+3)$.

За начальное приближение примем $x_o=0$, остальные приближения будем определять из $x_{i+1}=0{,}5-0{,}5\cdot\lg(2x_i+3)$.

i	$\mathbf{x_i}$	$\boldsymbol{\varphi(x)=x_{i+1}}$
0	0	0,2614
1	0.2614	0.2266
2	0.2266	0.2309
3	0.2309	0.2303
4	0.2303	0.2304
5	0.2304	

$x\approx 0{,}230$

Блок-схема метода



ЧИСЛЕННЫЕ МЕТОДЫ РЕШЕНИЙ СИСТЕМ НЕЛИНЕЙНЫХ УРАВНЕНИЙ

Пусть для вычисления неизвестных $x_1, x_2, \ldots, x_n$ требуется решить систему n нелинейных уравнений:

$$\begin{cases} f_1(x_1, x_2, \ldots x_n) = 0 \\ f_2(x_1, x_2, \ldots x_n) = 0 \\ \ldots \\ f_n(x_1, x_2, \ldots x_n) = 0 \end{cases}$$

Метод Ньютона-Рафсона

Метод базируется на разложении функций $f_j(x_1, x_2, \ldots, x_n)$, $j=1,2,\ldots,n$ в ряд Тейлора с отбрасыванием всех нелинейных членов разложения (т.е. функции линеаризуют). Затем осуществляется переход по всем переменным из текущей точки x^i в следующую x^{i+1}, которую считают решением, т.е. находят $h^i = x^{i+1} - x^i$, поэтому полагают $f_j(x^{i+1}) = 0$ $(j=1,2\ldots n)$. Получается система линейных уравнений относительно приращений h_j^i $(j=1,2\ldots n)$:

$$\begin{cases} f_1(x_1^i, x_2^i, \ldots x_n^i) + \dfrac{\partial f_1}{\partial x_1}(x_1^i, x_2^i, \ldots x_n^i) \bullet h_1^i + \dfrac{\partial f_1}{\partial x_2}(x_1^i, x_2^i, \ldots x_n^i) \bullet h_2^i + \ldots \dfrac{\partial f_1}{\partial x_n}(x_1^i, x_2^i, \ldots x_n^i) \bullet h_n^i = 0 \\ \quad \ldots \\ f_n(x_1^i, x_2^i, \ldots x_n^i) + \dfrac{\partial f_n}{\partial x_1}(x_1^i, x_2^i, \ldots x_n^i) \bullet h_1^i + \dfrac{\partial f_n}{\partial x_2}(x_1^i, x_2^i, \ldots x_n^i) \bullet h_2^i + \ldots \dfrac{\partial f_n}{\partial x_n}(x_1^i, x_2^i, \ldots x_n^i) \bullet h_n^i = 0 \end{cases},$$

ЧИСЛЕННЫЕ МЕТОДЫ РЕШЕНИЙ СИСТЕМ НЕЛИНЕЙНЫХ УРАВНЕНИЙ

Пусть для вычисления неизвестных $x_1, x_2, \ldots, x_n$ требуется решить систему n нелинейных уравнений:

$$\begin{cases} f_1(x_1, x_2, \ldots x_n) = 0 \\ f_2(x_1, x_2, \ldots x_n) = 0 \\ \ldots \\ f_n(x_1, x_2, \ldots x_n) = 0 \end{cases}$$

Метод Ньютона-Рафсона

Метод базируется на разложении функций $f_j(x_1, x_2, \ldots, x_n)$, $j=1,2,\ldots,n$ в ряд Тейлора с отбрасыванием всех нелинейных членов разложения (т.е. функции линеаризуют). Затем осуществляется переход по всем переменным из текущей точки x^i в следующую x^{i+1}, которую считают решением, т.е. находят $h^i = x^{i+1} - x^i$, поэтому полагают $f_j(x^{i+1}) = 0$ $(j=1,2\ldots n)$. Получается система линейных уравнений относительно приращений h_j^i $(j=1,2\ldots n)$:

$$\begin{cases} f_1(x_1^i, x_2^i, \ldots x_n^i) + \dfrac{\partial f_1}{\partial x_1}(x_1^i, x_2^i, \ldots x_n^i) \bullet h_1^i + \dfrac{\partial f_1}{\partial x_2}(x_1^i, x_2^i, \ldots x_n^i) \bullet h_2^i + \ldots \dfrac{\partial f_1}{\partial x_n}(x_1^i, x_2^i, \ldots x_n^i) \bullet h_n^i = 0 \\ \ldots \\ f_n(x_1^i, x_2^i, \ldots x_n^i) + \dfrac{\partial f_n}{\partial x_1}(x_1^i, x_2^i, \ldots x_n^i) \bullet h_1^i + \dfrac{\partial f_n}{\partial x_2}(x_1^i, x_2^i, \ldots x_n^i) \bullet h_2^i + \ldots \dfrac{\partial f_n}{\partial x_n}(x_1^i, x_2^i, \ldots x_n^i) \bullet h_n^i = 0 \end{cases},$$

которую решают. С найденными приращениями h_j переходят в новую точку по всем переменным $x_j^{i+1}=x_j^i+h_j^i$ (j=1,2,…n). Однако вследствие линейного приближения новая точка x^{i+1} не является решением, т.е. $f_j(x^{i+1})\neq 0$ (j=1,2,…n), ее считают следующим приближением и повторяют всю процедуру до тех пор, пока итерационный процесс не сойдется при удачно выбранном начальном приближении. Причем

$$x^{i+1} = x^i - W^{-1}(x^i)\cdot f(x^i),\text{ где}$$

$$x = \begin{pmatrix} x_1 \\ x_2 \\ \dots \\ x_n \end{pmatrix},\ \text{матрица аргументов}$$

$$f = \begin{pmatrix} f_1 \\ f_2 \\ \dots \\ f_n \end{pmatrix},\ \text{матрица функций}$$

W(x)–матрица Якоби системы функций $f_1,f_2,\dots,f_n$ относительно переменных $x_1,x_2,\dots,x_n$

$$W(x) = \begin{pmatrix} \frac{\partial f_1}{\partial x_1} & \frac{\partial f_1}{\partial x_2} & \dots & \frac{\partial f_1}{\partial x_n} \\ \frac{\partial f_2}{\partial x_1} & \frac{\partial f_2}{\partial x_2} & \dots & \frac{\partial f_2}{\partial x_n} \\ & \dots & & \\ \frac{\partial f_n}{\partial x_1} & \frac{\partial f_n}{\partial x_2} & \dots & \frac{\partial f_n}{\partial x_n} \end{pmatrix}$$

$W^{-1}(x)$-обратная матрица. Причем определитель матрицы –якобиан J – должен быть $\neq 0$ на каждой итерации.

Окончание поиска решения осуществляется по условию малости шагов по всем переменным одновременно, например, при выполнении условия $|h_j^i| \leq \xi$ j=1,2,…n, ξ–заданная погрешность. Можно также использовать условие отклонений функций $f(x_1,x_2,\dots,x_n)$ от нуля, например $\max\{f_j(x^i)\} \leq \xi$, или потребовать выполнения обоих условий одновременно.

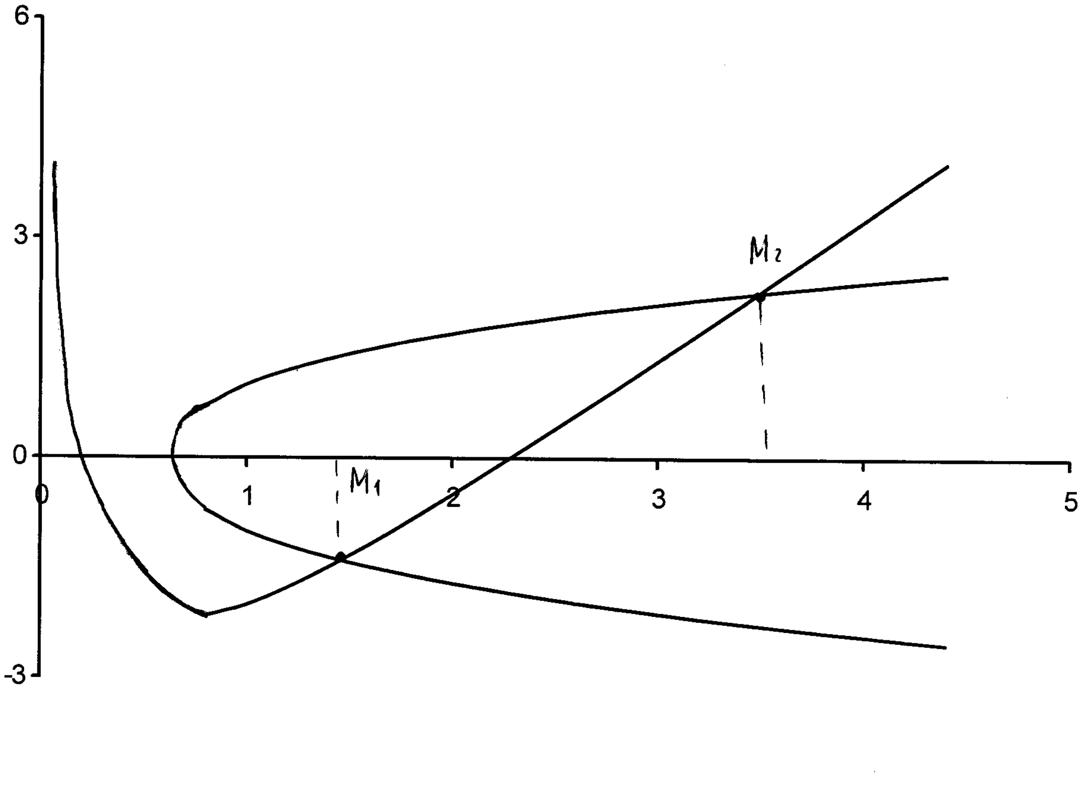
Пример:

Найти положительные решения системы с точностью ξ=0,001

$$\begin{cases} f_1(x_1,x_2) \equiv x_1 + 3 \bullet \lg x_1 - x_2^2 = 0 \\ f_2(x_1,x_2) \equiv 2x_1^2 - x_1x_2 - 5x_1 + 1 = 0 \end{cases}$$

Решение:

Отделим корни графически.



Кривые системы пересекаются приблизительно в точках M_1 и $M_2(3,4;2,2)$. Начальное приближение

$$x^{(0)}=\begin{pmatrix}3.4\\2.2\end{pmatrix}$$

Вычислим второе приближение, полагая

$$f(x)=\begin{pmatrix}f_1(x_1,x_2)\\f_2(x_1,x_2)\end{pmatrix}$$

имеем

$$f(x^{(0)})=\begin{pmatrix}3.4+3\lg 3.4-2.2^2\\2\bullet 3.4^2-3.4\bullet 2.2-5\bullet 3.4+1\end{pmatrix}=\begin{pmatrix}0.1544\\-0.3600\end{pmatrix}.$$

Составим матрицу Якоби

$$W(x)=\begin{pmatrix}\frac{\partial f_1}{\partial x_1}&\frac{\partial f_1}{\partial x_2}\\\frac{\partial f_2}{\partial x_1}&\frac{\partial f_2}{\partial x_2}\end{pmatrix}=\begin{pmatrix}1+\frac{3\bullet 0.43429}{x_1}&-2x_2\\4x_1-x_2-5&-x_1\end{pmatrix}$$

отсюда

$$W(x^{(0)})=\begin{pmatrix}1+\frac{3\bullet 0.43429}{3.4}&-2\bullet 2.2\\4\bullet 3.4-2.2-5&-3.4\end{pmatrix}$$

причем якобиан

$$J=\det W(x^{o})=23{,}4571\neq 0.$$

Обратная матрица

$$W^{-1}(x^{(0)})=\frac{1}{J}\begin{pmatrix}-3.4&4.4\\-6.4&1.3832\end{pmatrix}$$

Следующее приближение:

$$x^{(1)}=\begin{pmatrix}3.4\\2.2\end{pmatrix}-\frac{1}{23.4571}\begin{pmatrix}-3.4 & 4.4\\-6.4 & 1.3832\end{pmatrix}\begin{pmatrix}0.1500\\-0.3600\end{pmatrix}=\begin{pmatrix}3.4\\2.2\end{pmatrix}-\frac{1}{23.4571}\begin{pmatrix}-2.10896\\-1.48604\end{pmatrix}$$
$$=\begin{pmatrix}3.4\\2.2\end{pmatrix}+\begin{pmatrix}0.0899\\0.0633\end{pmatrix}=\begin{pmatrix}3.4899\\2.2633\end{pmatrix}$$

Аналогично для $x^{(2)}$, $x^{(3)}$…

i	**x_1**	**h_1**	**x_2**	**h_2**
0	3.4	0.0899	2.2	0.0633
1	3.4899	-0.0008	2.2633	-0,0012
2	3.4891	-0.0016	2.2621	-0.0005
3	3.4875		2.2616	

Останавливаемся на $x^{(3)}$

$x_1 \approx 3.4875$

$x_2 \approx 2.2616$

причем

$$f(x^{(3)})=\begin{pmatrix}0.0002\\0.0000\end{pmatrix}$$

Метод итераций

В этом методе предварительно все уравнения приводят к виду x=φ(x); т.е.

$$\begin{cases} x_1 = \varphi_1(x_1, x_2, \dots x_n) \\ x_2 = \varphi_2(x_1, x_2, \dots x_n) \\ \dots \\ x_n = \varphi_2(x_1, x_2, \dots x_n) \end{cases}$$

Берется произвольное начальное значение $x^{(o)}$ $(x_1^{(o)}, x_2^{(o)}, \dots, x_n^{(o)})$ и подставляется в уравнение. Из каждого уравнения системы находятся новые значения $x^{(1)}$ $(x_1^{(1)}, x_2^{(1)}, \dots, x_n^{(1)})$ и т.д. При выполнении условий

$$\sum_{i=1}^{n} M_{ki} < 1 \text{ или } \sum_{k=1}^{n} M_{ki} < 1, \text{ где } M_{ki} = \max\left|\frac{\partial \varphi_k}{\partial x_i}\right|$$

последовательность $x^{(0)}, x^{(1)}, x^{(2)} \dots$ сходится к решению.

Для обеспечения сходимости метода можно использовать следующий прием преобразования исходной системы к виду, удобному для итераций:

пусть

$x=(x_1,x_2,\dots,x_n)$ - матрица-строка аргументов

$\varphi(x)=(\varphi_1(x),\varphi_2(x),\dots\varphi_n(x))$ матрица-строка функций $\varphi(x)$, тогда система уравнений примет вид

$x=\varphi(x)$

Перепишем систему в виде:

$x^{(i+1)}=x^{(i)}+\Lambda f(x)=\varphi(x)$, где матрица Λ определяется как

$\Lambda= -(f'(x^{(0)}))^{-1}$, причем $\det f'(x^{(0)})\neq 0$.

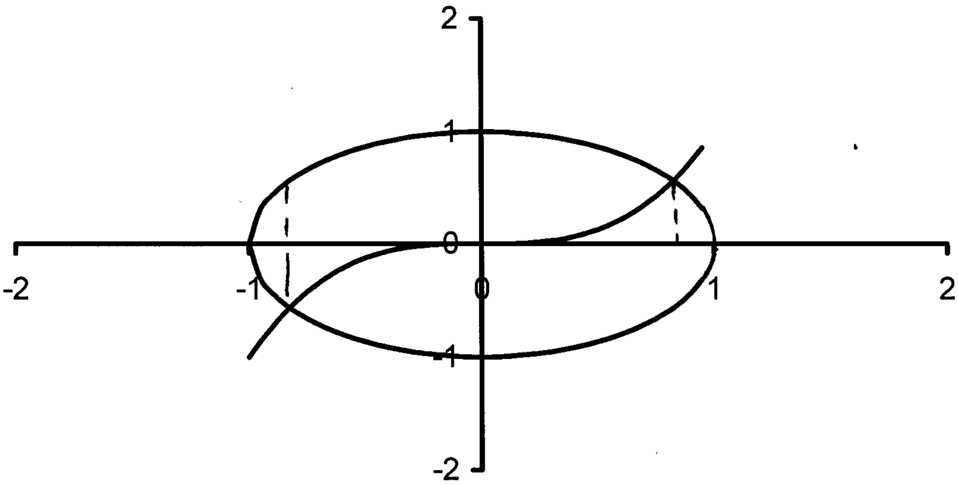
Пример:

Методом итерации решить систему

$$\begin{cases} x_1^2 + x_2^2 = 1 \\ x_1^3 - x_2 = 0 \end{cases}$$

Решение:

Отделим корни графически.



Система имеет два решения, отличающихся знаком. Найдем положительное решение системы. Примем

$$x^{(0)} = \begin{pmatrix} 0.9 \\ 0.5 \end{pmatrix}$$

Полагая $f(x) = \begin{pmatrix} x_1^2 + x_2^2 - 1 \\ x_1^3 - x_2 \end{pmatrix}$, имеем $f'(x) = \begin{pmatrix} 2x_1 & 2x_2 \\ 3x_1^2 & -1 \end{pmatrix}$

Отсюда $f'(x^{(0)}) = \begin{pmatrix} 1.8 & 1 \\ 2.43 & -1 \end{pmatrix}$

det $f'(x^{(0)})$=-1,8-2,43=-4,23≠0.

Обратная матрица

$$(f'(x^{(0)}))^{-1} = -\frac{1}{4.23}\begin{pmatrix} -1 & -1 \\ -2.43 & 1.8 \end{pmatrix}$$

т.о.

$$\Lambda = -(f'(x^{(0)}))^{-1} = \frac{1}{4.23}\begin{pmatrix} -1 & -1 \\ -2.43 & 1.8 \end{pmatrix}$$

Т.к.

$$\varphi(x) = x + \Lambda f(x) = \begin{pmatrix} x_1 \\ x_2 \end{pmatrix} - \frac{1}{4.23}\begin{pmatrix} 1 & 1 \\ 2.43 & -1.8 \end{pmatrix}\begin{pmatrix} x_1^2 + x_2^2 - 1 \\ x_1^3 - x_2 \end{pmatrix}, \text{то}$$

$$\mathrm{x}^{(1)} = \begin{pmatrix} x_1^{(0)} \\ x_2^{(0)} \end{pmatrix} - \frac{1}{4.23}\begin{pmatrix} 1 & 1 \\ 2.43 & -1.8 \end{pmatrix}\begin{pmatrix} x_1^{(0)2} + x_2^{(0)2} - 1 \\ x_1^{(0)3} - x_2^{(0)} \end{pmatrix} = \begin{pmatrix} 0.9 \\ 0.5 \end{pmatrix} - \frac{1}{4.23}\begin{pmatrix} 1 & 1 \\ 2.43 & -1.8 \end{pmatrix}\begin{pmatrix} 0.060 \\ 0.229 \end{pmatrix} =$$

$$\begin{pmatrix} 0.9 \\ 0.5 \end{pmatrix} - \begin{pmatrix} 0.0683 \\ -0.0630 \end{pmatrix} = \begin{pmatrix} 0.8317 \\ 0.5630 \end{pmatrix};$$

$$x^{(2)} = \begin{pmatrix} 0.8317 \\ 0.5630 \end{pmatrix} - \frac{1}{4.23}\begin{pmatrix} 1 & 1 \\ 2.43 & -1.8 \end{pmatrix}\begin{pmatrix} 0.8317^2 + 0.5630^2 - 1 \\ 0.8317^3 - 0.5630 \end{pmatrix} = \begin{pmatrix} 0.8268 \\ 0.5633 \end{pmatrix}$$

$$x^{(3)} = \begin{pmatrix} 0.8268 \\ 0.5633 \end{pmatrix} - \begin{pmatrix} 0.0007 \\ 0.0002 \end{pmatrix} = \begin{pmatrix} 0.8261 \\ 0.5631 \end{pmatrix}$$

$$\mathrm{x}^{(4)} = \begin{pmatrix} 0.8261 \\ 0.5631 \end{pmatrix} - \begin{pmatrix} 0.0000 \\ -0.0005 \end{pmatrix} = \begin{pmatrix} 0.8261 \\ 0.5636 \end{pmatrix}$$

Ограничиваясь $x^{(4)}$, $x_1 \approx 0{,}8261$, $x_2 \approx 0{,}5636$, причем $f(x) = \begin{pmatrix} 0.0000 \\ 0.0002 \end{pmatrix}$ с точностью 10^{-4}.

МЕТОДЫ ОБРАБОТКИ ТАБЛИЧНЫХ ДАННЫХ

Пусть величина **y** является функцией аргумента **x**. Это означает, что любому значению **x** из области определения поставлено в соответствие значение **y**. На практике часто неизвестна явная связь между **y** и **x**, т.е. невозможно записать эту связь в виде некоторой зависимости y=f(x). В других случаях даже при известной зависимости y=f(x) она настолько сложна (например, содержит трудно вычисляемые выражения, интегралы, и т.п.), что ее использование в практических целях затруднительно.

Пусть дана таблица значений $\{x_i, y_i\}$, $i=0,1,\ldots n$ и необходимо получить значение величины **y** в других точках, отличных от $\mathbf{x_i}$, причем точная связь y=f(x) неизвестна. Для этой цели служит задача о приближении (аппроксимации) функции: данную функцию f(x) требуется приближенно заменить (аппроксимировать) некоторой функцией φ(x) так, чтобы отклонение φ(x) от f(x) в заданной области было наименьшим. Функция φ(x) при этом называется аппроксимирующей.

Если приближение строится на заданном дискретном множестве точек $\{x_i\}$, то аппроксимация называется точечной. При построении приближения на непрерывном множестве точек аппроксимация называется непрерывной, или интегральной.

Интерполяция

Одним из основных типов точечной аппроксимации является интерполяция – нахождение значения таблично заданной функции в тех точках внутри данного интервала, где она не задана. Если функция восстанавливается в точках за пределами заданного интервала, то используют термин экстраполяция.

При построении интерполяционной функции φ(x) проходящей через все заданные точки – узлы интерполяции – возникают три проблемы:

- выбора интерполяционной функции φ(x);
- оценка погрешности R(x);

- размещение узлов интерполяции для обеспечения возможной наивысшей точности восстановления функции.

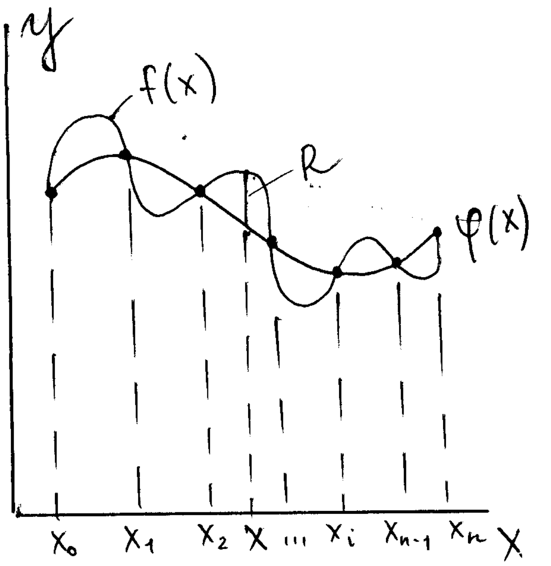


Рис. 7

Чаще всего интерполяционную функцию представляют в виде полинома, принимающего в заданных точках x_i те же значения y_i, что и функция f(x), т.е. $\varphi(x_i)=y_i$, $i=0,1,\ldots n$. Точки x_i называются узлами интерполяции, φ(x)- интерполяционный многочлен

$$\varphi(x)=a_o+a_1x+a_2x^2+\ldots+a_nx^n.$$

Метод Лагранжа

Пусть известны значения некоторой функции f(x) в n+1 произвольных различных точках $y_i=f(x_i)$, $i=0,1\ldots n$. Для интерполирования (восстановления) функции в какой-либо точке $x\in[x_o,x_n]$ необходимо построить интерполяционный полином n-го порядка:

$$L_n(x)=y_oQ_o(x)+\ldots+y_jQ_j(x)+\ldots+y_nQ_n(x),\text{ где}$$

$$Q_j(x)=\frac{(x-x_0)...(x-x_{j-1})(x-x_{j+1})...(x-x_n)}{(x_j-x_0)...(x_j-x_{j-1})...(x_j-x_n)}$$

Оценить погрешность интерполяции в точке $x\in[x_o,x_n]$ можно по формуле:

$$R(x)=|f(x)-L_n(x)|\leq\frac{M_{n+1}}{(n+1)!}\prod_{i=0}^{n}|(x-x_i)|,$$ где

$M_{n+1}=\max|f^{(n+1)}(x)|$ –максимальное значение (n+1)-ой производной исходной функции f(x) на отрезке $[x_o,x_n]$.

Пример:

Дана таблично заданная функция:

x	0	1	2	6
y	-1	-3	3	1187

Требуется найти y при x=4.

Решение:

n=3

$$L_3(x)=y_0\frac{(x-x_1)(x-x_2)(x-x_3)}{(x_0-x_1)(x_0-x_2)(x_0-x_3)}+y_1\frac{(x-x_0)(x-x_2)(x-x_3)}{(x_1-x_0)(x_1-x_2)(x_1-x_3)}+$$
$$+y_2\frac{(x-x_0)(x-x_2)(x-x_3)}{(x_2-x_0)(x_2-x_1)(x_2-x_3)}+y_3\frac{(x-x_0)(x-x_1)(x-x_2)}{(x_3-x_0)(x_3-x_1)(x_3-x_2)}=$$
$$-1\frac{(4-1)(4-2)(4-6)}{(0-1)(0-2)(0-6)}-3\frac{(4-0)(4-2)(4-6)}{(1-0)(1-2)(1-6)}+3\frac{(4-0)(4-1)(4-6)}{(2-0)(2-1)(2-6)}$$
$$+1187\frac{(4-0)(4-1)(4-2)}{(6-0)(6-1)(6-2)}=255$$

Т.к. отсутствует информация о значении четвертой производной исходной функции, то оценить погрешность сложно.

Пример:

Дана функция y=$\sqrt{x}$ (для иллюстрации метода считаем ее “сложной”). Найти погрешность интерполяции функции при x=115.

Решение:

Знаем, что

x	100	121	144
y	10	11	12

Всего три узла интерполяции, n=2. Оценим максимальное значение третьей производной

$$y' = \frac{x^{-1/2}}{2}, \quad y'' = -\frac{x^{-3/2}}{4}, y''' = 3\frac{x^{-5/2}}{8}$$

$$M_3 = \max|y'''| = 3\frac{(100)^{-5/2}}{8}$$

Погрешность при интерполяции по трем узлам:

$$R \le \frac{3 \bullet 100^{-5/2} \bullet |(115-100)(115-121)(115-144)|}{8 \bullet 3!} \approx 1.6 \bullet 10^{-3}$$

Метод Ньютона

Пусть известны значения некоторой функции f(x) в n+1 произвольных, попарно не совпадающих точках y=f(x), i=0,…,n. В общем случае интерполяция по формулам Ньютона может производиться для произвольно расположенных точек – узлов интерполяции, но чаще применяется для равномерно расположенных.

Рассмотрим случай с равномерным расположением узлов

$x_{i+1}=x_i+h$, где $h=(x_o-x_n)/n$.

ЗАМЕЧАНИЕ

Пусть $y=f(x)$–заданная функция. Обозначим через $\Delta x=h$ фиксированную величину аргумента (шаг), тогда выражение

$$\Delta y \equiv \Delta f(x)=f(x+\Delta x)-f(x)$$

называется первой конечной разностью функции y. Аналогично определяются конечные разности высших порядков.

$$\Delta^n y=\Delta(\Delta^{n-1}y)\ ,\ (n=2,3,\dots)$$

Например,

$$\Delta^2 y=\Delta[f(x+\Delta x)-f(x)]=[f(x+2\Delta x)-f(x+\Delta x)]-[f(x+\Delta x)-f(x)].$$

Пример:

Построить конечные разности для функции $P(x)=x^3$, считая шаг $\Delta x=1$.

Решение:

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

$\Delta^2 P(x)=[3(x+1)^2+3(x+1)+1]-(3x^2+3x+1)=6x+6$

$\Delta^3 P(x)=[6(x+1)+6]-(6x+6)=6$

$\Delta^n P(x)=0,\ n>3.$

Интерполяционный многочлен Ньютона имеет вид:

$$P_n(x)=y_0+\frac{\Delta y_0(x-x_0)}{h}+\frac{\Delta^2 y_0(x-x_0)(x-x_1)}{2!h^2}+\frac{\Delta^3 y_0(x-x_0)(x-x_1)(x-x_2)}{3!h^3}+\dots+$$
$$+\frac{\Delta^n y_0(x-x_0)(x-x_1)\dots(x-x_{n-1})}{n!h^n}$$

Часто вводят безразмерную величину q, показывающую, сколько содержится шагов от x_o до заданной точки x:

$$q=(x-x_o)/h:$$

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

$$+ \frac{\Delta^n y_0 q(q-1)\ldots(q-n+1)}{n!}$$

Обе формулы называются первой интерполяционной формулой Ньютона и применяются при интерполяции вперед (в сторону увеличения x) или при экстраполяции назад (левее x_o).

Второй интерполяционный многочлен Ньютона применяется при интерполяции назад (т.е. в конце интервала, но левее x_n) или при экстраполяции вперед (правее x_n):

$$P_n(x) = y_n + \Delta y_{n-1} q + \frac{\Delta^2 y_{n-2} q(q+1)}{2!} + \frac{\Delta^3 y_{n-3} q(q+1)(q+2)}{3!} + \ldots +$$

$$+ \frac{\Delta^n y_0 q(q+1)\ldots(q+n-1)}{n!}$$

Оценить погрешность интерполяции можно по формуле:

$$R(x) = |f(x) - P_n(x)| \le \frac{\max\{|\Delta^{(n+1)} y(x)|\}}{(n+1)!} \bullet |q(q-1)(q-2)\ldots(q-n)|$$

Пример:

Даны следующие точки

x	2.0	2.1	2.2	2.3	2.4	2.5	2.6
y	0.0540	0.0440	0.0355	0.0283	0.0224	0.0175	0.0136

Необходимо найти y(2,05).

Решение:

Используем для интерполяции первые три точки, остальные используем для оценки погрешности.

Т.о. $n=2$, $h=0,1$

$Q=(x-x_o)/h=0,5$

Составим таблицу конечных разностей и воспользуемся интерполяционной формулой Ньютона:

x	y	Δy	$\Delta^2 y$	$\Delta^3 y$
2.0	0.0540	-0.0100	0.0015	-0.0002
2.1	0.0440	-0.0085	0.0013	0
2.2	0.0355	-0.0072	0.0013	-0.0003
2.3	0.0283	-0.0059	0.0010	-0.0010
2.4	0.0224	-0.0049	0	
2.5	0.0175	-0.0049		
2.6	0.0136			

$$y(2.05)=0.0540+0.5(-0.01)+0.5(0.5-1)\bullet\frac{0.0015}{2!}=0.0488125.$$

Оценим погрешность найденного у:

$\max\{|\Delta^3 y(x)|\}=0,0010$

$$R\leq\frac{0.001\bullet|0.5(0.5-1)(0.5-2)|}{3!}=0.0000625$$

Метод сплайнов

Часто возникает задача восстановления не только значений функции, но и ее первой и второй производной. Для решения такой задачи применяется сплайновая интерполяция.

Сплайн–это функция, которая на каждом межузловом интервале совпадает с некоторым полиномом, своим для каждого интервала. Полиномы соседних интервалов стыкуются так, чтобы функция была непрерывной. Дополнительно требуется непрерывность нескольких производных.

Наибольшее распространение получила интерполяция кубическими сплайнами. Кубический сплайн склеивается из полиномов третьей степени, которые для i-го участка записываются

$$y=a_i(x-x_i)^3+b_i(x-x_i)^2+c_i(x-x_i)+d_i$$

Для всего интервала будет n кубических полиномов, отличающихся коэффициентами a_i,b_i,c_i,d_i. Чаще всего узлы при сплайновой интерполяции располагаются равномерно, $x_{i+1}-x_i=h=const$.

Т.о. необходимо найти четыре коэффициента при условии прохождения каждого полинома через две точки (x_i,y_i) и (x_{i+1},y_{i+1})

$$y_i=d \qquad i=0,1,2,\ldots n-1$$

$$y_{i+1}=a_ih^3+b_ih^2+c_ih+d_i \qquad i=0,1,2,\ldots n-1$$

Первое условие соответствует прохождению полинома через начальную точку, второе–через конечную точку.

Найти все коэффициенты нельзя, т.к. уравнений больше, чем неизвестных. Дополним уравнения условием гладкости функции (т.е. непрерывности первой производной) и гладкости первой производной (т.е. непрерывности второй производной) в узлах интерполяции. Математически эти условия записываются как равенства первой и второй производных в конце i-го и в начале (i+1)-го участков.

Т.к.

$$y'=3a_i(x-x_i)^2+2b_i(x-x_i)+c_i$$
$$y''=6a_i(x-x_i)+2b_i,$$

то

$$3a_ih^2+2b_ih+c_i=c_{i+1}, \quad i=0,1,2,\ldots,n-2.$$
$$6a_ih+2b_i=2b_{i+1}, \quad i=0,1,2,\ldots,n-2.$$

Получают систему линейных уравнений для всех участков, содержащую 4n-2 уравнения с 4n неизвестными (a_1, a_2, $\ldots a_n$, b_1, $\ldots d_n$ – коэффициенты сплайнов). Для решения системы добавляют два граничных условия одного из следующих видов:

1) $y''(x_o)=y''(x_n)=0$

2) $y'(x_o)=g_o$, $y'(x_n)=g_n$

3) $y'(x_o)=y'(x_n)=0$

4) $y'(x_o)=y'(x_n)$, $y''(x_o)=y''(x_n)$

Совместное решение 4n уравнений позволяет найти все 4n коэффициента.

Среднеквадратичное приближение

При большом количестве узлов интерполяции получается высокая степень многочлена $\varphi(x)=a_o+a_1x+\ldots+a_nx^n$. Кроме того, табличные данные могли быть получены путем измерений и содержать ошибки. Построение аппроксимирующего многочлена с условием обязательного прохождения его графика через эти экспериментальные точки означало бы повторение допущенных при измерениях ошибок. В этом случае строится аппроксимирующая функция, в целом наиболее близко проходящая около данных точек.

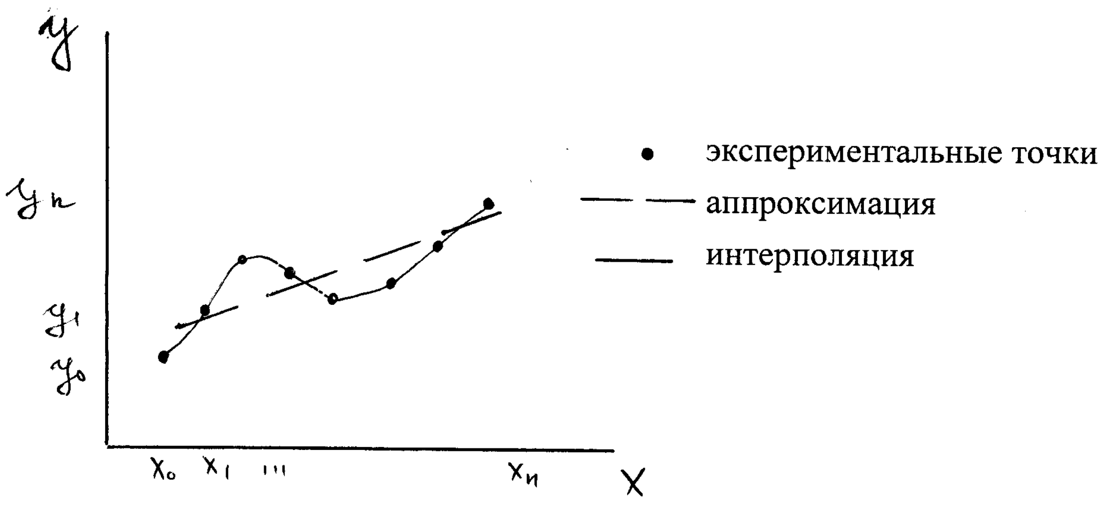


Рис.8

Критерием близости исходной и аппроксимирующей функций при среднеквадратичном приближении является минимальное значение суммы квадратов отклонений между значениями аппроксимирующего многочлена и заданных таблично

$$R=\sum_{i=0}^{n}\beta_i[\varphi(x_i)-f(x_i)]^2,$$ где

β-весовые коэффициенты, учитывающие важность i-той точки.

При среднеквадратичном приближении степень аппроксимирующего полинома m меньше n:

$$\varphi(x)=a_o+a_1x+\ldots+a_mx^m \ (m<n).$$

На практике стараются подобрать аппроксимирующий многочлен как можно меньшей степени (m=1,2,3). При построении аппроксимирующего полинома подбираются коэффициенты $a_o,a_1,\ldots,a_m$ т.о., чтобы величина R была наименьшей – это т.н. метод наименьших квадратов (МНК).

ЗАМЕЧАНИЕ

В теории вероятностей доказывается, что полученные таким методом значения параметров $a_o,a_1,\ldots a_m$ наиболее вероятны, если отклонения $\varphi(x_i)$-$f(x_i)$ подчиняются нормальному закону распределения.

Коэффициенты $a_o,a_1,\ldots,a_m$ находят из необходимого условия минимума функции R, для которой эти коэффициенты являются независимыми переменными (β=1):

$$\begin{cases} \dfrac{\partial R}{\partial a_0} = 0 \\ \dfrac{\partial R}{\partial a_1} = 0 \\ \ldots \\ \dfrac{\partial R}{\partial a_m} = 0 \end{cases} \qquad \begin{cases} \dfrac{\partial R}{\partial a_0} = 2\sum\limits_{i=0}^{n}(a_0 + a_1 x_i + \ldots a_m x_i^m - f(x_i)) = 0 \\ \dfrac{\partial R}{\partial a_1} = 2\sum\limits_{i=0}^{n}(a_0 + a_1 x_i + \ldots a_m x_i^m - f(x_i)) \bullet x_i = 0 \\ \ldots \\ \dfrac{\partial R}{\partial a_m} = 2\sum\limits_{i=0}^{n}(a_0 + a_1 x_i + \ldots a_m x_i^m - f(x_i)) \bullet x_i^m = 0 \end{cases}$$

После сокращения на два, раскрытия скобок и изменения порядка суммирования получим т.н. систему нормальных уравнений, из решения которой находим параметры $a_o,a_1,\ldots a_m$:

$$\begin{cases} (n+1)a_0 + a_1\sum\limits_{i=0}^{n} x_i + a_2\sum\limits_{i=0}^{n} x_i^2 + \ldots a_m\sum\limits_{i=0}^{n} x_i^m = \sum\limits_{i=0}^{n} y_i \\ a_0\sum\limits_{i=0}^{n} x_i + a_1\sum\limits_{i=0}^{n} x_i^2 + a_2\sum\limits_{i=0}^{n} x_i^3 + \ldots a_m\sum\limits_{i=0}^{n} x_i^{m+1} = \sum\limits_{i=0}^{n} x_i y_i \\ \ldots \\ a_0\sum\limits_{i=0}^{n} x_i^m + a_1\sum\limits_{i=0}^{n} x_i^{m+1} + a_2\sum\limits_{i=0}^{n} x_i^{m+2} + \ldots a_m\sum\limits_{i=0}^{n} x_i^{2m} = \sum\limits_{i=0}^{n} x_i^m y_i \end{cases}$$

Пример:

Необходимо найти аппроксимирующую функцию в виде линейного полинома $y=a_o+a_1x$ по имеющимся экспериментальным данным

x	-26	-22	-16	-11	-5	3	10	25	42
y	66.7	71.0	76.3	80.6	85.7	92.9	99.4	113.6	125.1

Решение:

Составим систему линейных уравнений относительно a_o и a_1:

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

$n=8$ $\Sigma x_i=0$ $\Sigma x_i^2=4060$ $\Sigma y_i=811.3$ $\Sigma y_ix_i=3534.8$

$$\begin{cases} 9a_0=811.3 \\ 4060a_1=3534.8 \end{cases} \qquad \begin{cases} a_0=90.1 \\ a_1=0.87 \end{cases}$$

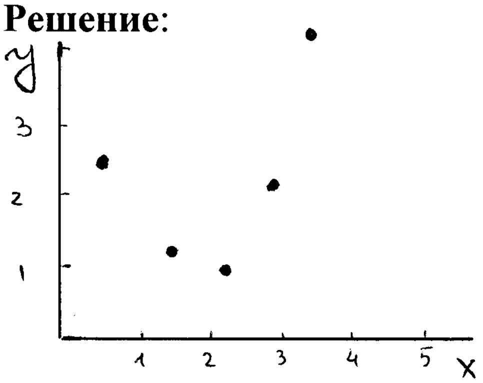
Аппроксимирующая функция $y=90,1+0,87x$.

Пример:

Используя метод МНК вывести эмпирическую формулу для функции $y=f(x)$, заданной в табличном виде:

x	0.75	1.50	2.25	3.00	3.37
y	2.50	1.20	1.12	2.25	4.28

Решение:



Изобразим табличные значения графически, откуда видно, что в качестве аппроксимирующей функции можно взять параболу: $\varphi(x)=a_0+a_1x+a_2x^2$, $m=2$ $n=4$

Система нормальных уравнений имеет вид:

$$\begin{cases} 5a_0 + a_1\sum_{i=0}^{n} x_i + a_2\sum_{i=0}^{n} x_i^2 = \sum_{i=0}^{n} y_i \\ a_0\sum_{i=0}^{n} x_i + a_1\sum_{i=0}^{n} x_i^2 + a_2\sum_{i=0}^{n} x_i^3 = \sum_{i=0}^{n} x_i y_i \\ a_0\sum_{i=0}^{n} x_i^2 + a_1\sum_{i=0}^{n} x_i^3 + a_2\sum_{i=0}^{n} x_i^4 = \sum_{i=0}^{n} x_i^2 y_i \end{cases}$$

$\Sigma x_i=11{,}25$ $\Sigma x_i^2=30{,}34$ $\Sigma x_i^3=94{,}92$ $\Sigma x_i^4=309{,}76$ $\Sigma y_i=11{,}35$

$\Sigma x_i y_i=29{,}00$ $\Sigma x_i^2 y_i=90{,}21$.

Итого:

$$\begin{cases} 5a_0 + 11.25a_1 + 30.94a_2 = 11.35 \\ 11.25a_0 + 30.94a_1 + 94.92a_2 = 29.00 \\ 30.94a_0 + 94.92a_2 + 309{,}76a_2 = 90.21 \end{cases} \qquad \begin{cases} a_0 = 5.54 \\ a_1 = -4.73 \\ a_2 = 1.19 \end{cases}$$

$\varphi(x) = 5.54 - 4.73x + 1.19x^2$.

СПИСОК РЕКОМЕНДУЕМОЙ ЛИТЕРАТУРЫ

1. Д.Мак-Кракен, У.Дорн. Численные методы и программирование на Фортране. М.: Мир, 1977. 584 с.
2. К.Эберт, Х.Эдерер. Компьютеры. Применение в химии. М.: Мир, 1988. 416 с.
3. Л.И.Турчак. Основы численных методов. М.: Наука, 1987. 320 с.
4. Б.П.Демидович, И.А.Марон. Основы вычислительной математики. М.: Наука, 1966. 664 с.
5. Б.П.Демидович, И.А.Марон, Э.З.Шувалова. Численные методы анализа. М.: Наука, 1967. 368 с.
6. Х.Гулд, Я.Тобочник. Компьютерное моделирование в физике. В 2-х частях. М.: Мир, 1990.
7. Вычислительная математика / Данилина Н.И., Дубровская Н.С., Кваша О.П., Смирнов Г.Л. М.: Высшая школа, 1985. 472 с.
8. Б.Банди. Методы оптимизации. Вводный курс. М.: Радио и связь, 1988. 128 с.
9. Ю.В.Васильков, Н.Н.Василькова. Компьютерные технологии вычислений в математическом моделировании. М.: Финансы и статистика, 1999. 256 с.