РЕШЕНИЕ УРАВНЕНИЙ МЕТОДОМ НЬЮТОНА

Цель работы: освоить методику численного решения нелинейных уравнений методом Ньютона, исследовать его свойства.

1. ТЕОРЕТИЧЕСКИЕ СВЕДЕНИЯ

1. 1. Метод Ньютона (касательных)

Метод Ньютона это метод решения нелинейных уравнений вида

$$f(x) = 0, \tag{1}$$

основанный на линейной аппроксимации:

$$f^*(x) = kx + b.$$

Коэффициенты k и b выбираются таким образом, чтобы в текущей точке x_i совпадали значения самих функций $f(x)$ и $f^*(x)$ а также их первых производных:

$$\begin{cases} f(x_i) = f^*(x_i) \\ f'(x_i) = f^{*\prime}(x_i). \end{cases}$$

Решив эту систему относительно k и b, можно записать уравнение приближенной функции в явном виде:

$$f^*(x) = f(x_i) + f'(x_i)(x - x_i)\,. \tag{2}$$

Поскольку функция $f^*(x)$ линейна, найти решение приближенной задачи легко:

$$f(x_i) + f'(x_i)(x_{i+1} - x_i) = 0,$$

$$x_{i+1} = x_i - \frac{f(x_i)}{f'(x_i)}\,. \tag{3}$$

Найденное значение и будет следующим, более точным приближением к решению. Формула (3) и лежит в основе метода Ньютона. Необходимым, но не достаточным условием для его применимости является требование гладкости целевой функции f(x): функция f(x) должна иметь производную.

Линейное приближение функции в точке имеет простую геометрическую интерпретацию. Действительно, уравнение (2) представляет собой уравнение касательной к функции $f(x)$ в точке x_i. Пересечение этой касательной с осью x дает следующее приближение. На Рис. 1 показаны два последовательных приближения.

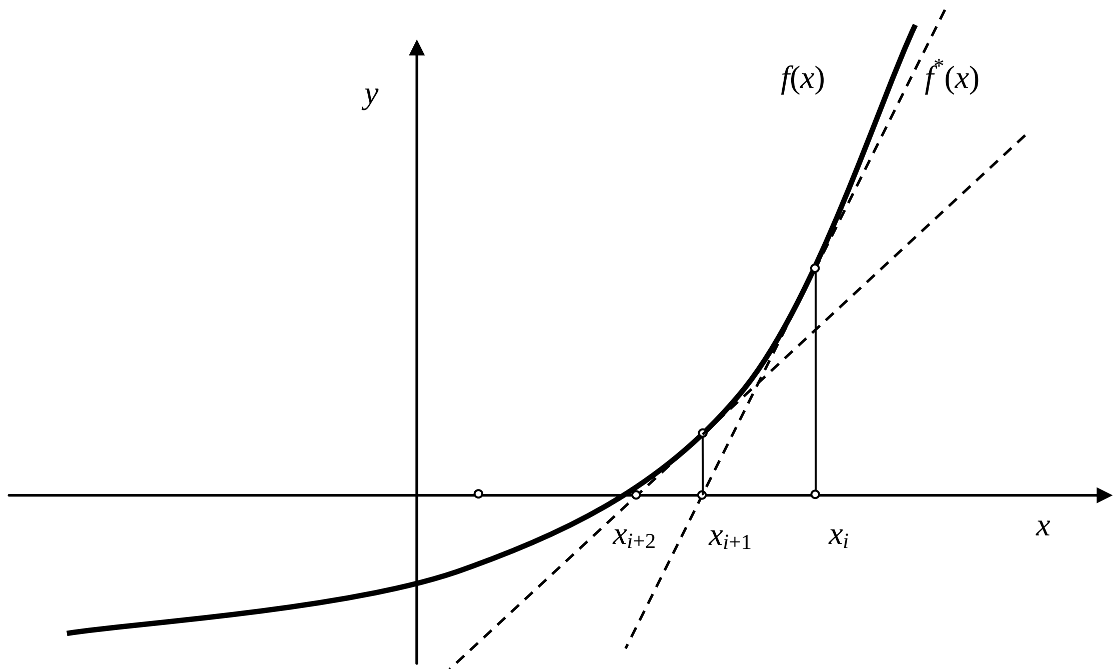

Рис. 1. Решение уравнений методом Ньютона

.

Алгоритм

1) Задается начальная точка x_0.

2) По формуле итерации вычисляется следующая точка: $x_{i+1} = x_i - \frac{f(x_i)}{f'(x_i)}$

3) Проверяется условие завершения, для этого обычно расстояние между последними двумя приближениями сравнивается с заранее заданной величиной погрешности: $|x_{i+1}-x_i| < \varepsilon$. Если условие выполнено, то алгоритм завершается, и текущая точка является ответом, иначе – возврат на шаг 2.

Условие завершения может меняться в зависимости от конкретной задачи. Обычно алгоритм останавливается, когда расстояние между соседними приближениями $|x_i-x_{i-1}|$ или абсолютная величина $|f(x_i)|$ становятся меньше некоторой заданной малой величины ε.

1.2. Скорость сходимости

Скорость сходимости итерационного метода это число, которое характеризует, насколько быстро уменьшается погрешность с каждой итерацией. Ее можно определить следующим образом: говорят, что численный метод имеет скорость сходимости α, если в пределе при бесконечном количестве итераций выполняется равенство:

$$r_{i+1} = kr_i^{\alpha},$$

где i – номер итерации, r_i - погрешность на итерации i, $k>0$ – произвольный коэффициент пропорциональности, значение которого несущественно.

Выделают несколько особых случаев скорости сходимости. Сходимость со скоростью $\alpha = 1$ называют **линейной сходимостью**. Это сравнительно медленная сходимость. В таких методах с каждым шагом погрешность уменьшается в некоторое примерно постоянное число раз. При этом количество значащих цифр ответа растет пропорционально числу итераций. Типичные представители методов с такой скоростью сходимости – изученные ранее методы деления пополам и золотого сечения.

Пример: зависимость величины погрешности от количества итераций в методе деления пополам

Номер итерации i	Погрешность r_i	Количество значащих цифр
0	1.00000	0
1	0.50000	0
2	0.25000	0
3	0.12500	0
4	0.06250	1
5	0.03125	1
6	0.01562	1
7	0.00781	2
8	0.00391	2
9	0.00195	2
10	0.00097	3

Скорость сходимости $\alpha = 2$ называют **квадратичной сходимостью**. У методов с такой сходимостью количество значащих цифр примерно удваивается с каждой итерацией.

Известно, что такой сходимостью обладает метод Ньютона, если в найденном корне производная целевой функции не равна нулю. Если же производная равна нулю (например, такое наблюдается в случае кратных корней у полиномиальных уравнений), скорость сходимости может падать до линейной.

Пример: уменьшение погрешности при решении методом ньютона уравнения $x^2 = 2$

Номер итерации i	Приближение x_i	Погрешность r_i	Количество значащих цифр
0	0.1	1.31	0
1	10.05	8.6	0
2	5.124502487562189	3.71	0
3	2.7573921384195743	1.34	0
4	**1.**741357580449592	0.33	1
5	**1.4**44943381958916	0.031	2
6	**1.414**540330128693	0.00033	4
7	**1.4142136**001158034	0.000000038	8
8	**1.414213562373095**6	$4.4 \cdot 10^{-16}$	16

1.2. Эмпирическая оценка скорости сходимости

Если скорость сходимости неизвестна, она может быть оценена эмпирически сладующим способом. Возьмем логарифм от правой и левой частей уравнения, определяющего скорость сходимости:

$$\log r_{i+1} = \log k + \alpha \log r_i .$$

Из полученного равенства видно, что зависимость логарифма погрешности на текущем шаге от логарифма погрешности на предыдущем шаге примерно линейная, причем скорость сходимости α это коэффициент данной зависимости.

Если построить эту зависимость на графике, то точки должны оказаться примерно на прямой линии. На рис.2 показаны точки, построенные по последнему примеру. Начальные точки, соответствующие большой погрешности, находятся в правом верхнем углу графика, конечные – в нижнем левом.

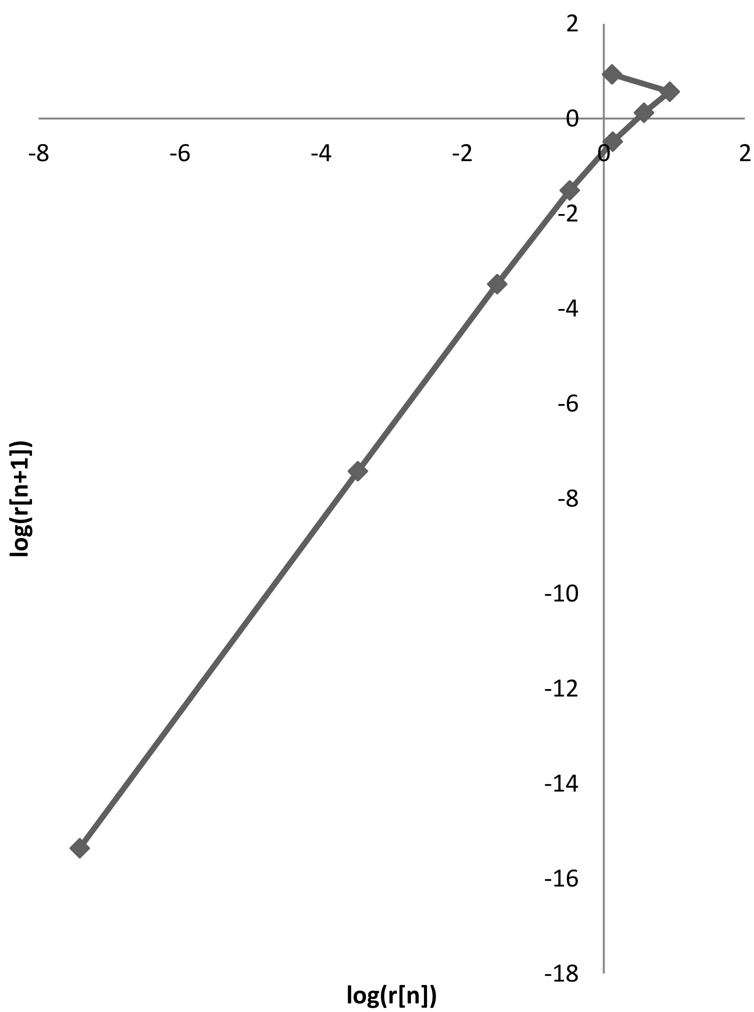

Рис. 2. Зависимость логарифма погрешности на текущем шаге (вертикальная ось) от его значения на предыдущем шаге (горизонтальная ось).

Несложно заметить, что после короткого периода хаотичного поведения в начале точки постепенно приближаются к прямой линии. Угловой коэффициент этой линии и равен скорости сходимости. Оценим его по двум последним точкам. Координаты этих точек равны: (log(0.00033); log(3.8e-8)) и (log(3.8e-8); log(4.4e-16)). Тогда угловой коэффициент равен:

$$\alpha = \Delta y / \Delta x = (\log(4.4 \cdot 10^{-16}) - \log(3.8 \cdot 10^{-8}) / (\log(3.8 \cdot 10^{-8}) - \log(0.00033)) = 2.01,$$

что весьма близко к теоретическому значению 2. Более точный результат может быть получен с использованием линейной регрессии.

2. ЗАДАНИЕ ПО РАБОТЕ

В лабораторной работе требуется на любом языке программирования реализовать решение двух заданных уравнений методом Ньютона. Уравнения указаны в варианте задания.

Целевая функция содержит два свободных параметра, значения которых также указаны в варианте задания.

В результате выполнения программа должна вывести таблицу из трех столбцов: номер итерации i, текущее приближение x_i, оценку погрешности $r_i = \left|x_i - x_{i-1}\right|$.

По полученным данным необходимо произвести эмпирическую оценку скорости сходимости метода.

3. СОДЕРЖАНИЕ ОТЧЕТА

Отчет по лабораторной работе должен содержать следующие элементы:

1. Постановка задания.
2. Данные варианта.
3. Аналитическое решение уравнения, решение должно быть приведено с точностью до 8 знаков после запятой.
4. Вычислить значение производной целевой функции в каждом найденном корне.
5. Эскиз графика целевой функции. Интервал построения графика должен быть выбран так, чтобы захватывать область корня и показывать характерные особенности функции.
6. Текст программы.
7. Результаты выполнения программы, оформленные в виде таблицы из 3 столбцов: итерация, текущее приближение, погрешность. Если число шагов велико, привести только последние 10 шагов. Если метод не сходится – выбрать начальную точку ближе к точному значению корня, вычисленному в пункте 3.
8. Оценку скорости сходимости (взять последние 3 итерации).
9. Выводы.

5. ВАРИАНТЫ ЗАДАНИЙ

№	Уравнение	Примечания	Значения свободных параметров A, B	
			A	**B**
1	x^2+A*x+B=0	Несколько корней	0	-2
2	A*e^x+B=0		1	-2
3	A^x+B=0		2	3
4	A^x-B*e^x=0		7	2
5	(x-1)*(x^2+A*x+B)=0	Несколько корней	1	2
6	(x-2)*(A^x+B)=0	Несколько корней	2	-4
7	(x^2+A*x+B)^2=0	Несколько корней	0	-2
8	(A*e^x+B)^2=0		1	-7
9	(A^x-B*e^x)^2=0		3	2
10	x^2+A*x+B=0	Несколько корней	1	-1
11	A*e^x+B=0		1	-4
12	A^x+B=0		3/2	-5
13	A^x-B*e^x=0		3/2	2
14	(x-1)*(x^2+A*x+B)=0	Несколько корней	2	-3
15	(x-2)*(A^x+B)=0	Несколько корней	3	-9
16	(x^2+A*x+B)^2=0	Несколько корней	1	-1
17	(A*e^x+B)^2=0		1	-3
18	(A^x-B*e^x)^2=0		4/3	2
19	A^x+B=0		5	-7
20	A^x-B*e^x=0		2	20
21	(x-1)*(x^2+A*x+B)=0	Несколько корней	3	-4

Точность, с которой необходимо искать решение уравнения, одинакова для всех вариантов и равна 10^{-14}.