1. Информация об используемых методах:

Метод секущих

Упрощение метода Ньютона, использующего производную.

Итерационный процесс имеет вид:

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

Порядок сходимости метода секущих ниже, чем у метода касательных и равен в случае однократного корня золотому сечению $\varphi = \frac{\sqrt{5}+1}{2} = 1{,}6180339887\ldots$

Повторять операцию следует до тех пор, пока $|x_i - x_{i-1}|$ не станет меньше или равно заданному значению погрешности.

Метод парабол (Мюллера)

Метод Мюллера развивает идею метода секущих, который строит на каждом шаге итерации прямые, проходящие через две точки на графике $y = f(x)$. Вместо этого метод Мюллера использует три точки, строит параболу, проходящую через эти три точки, и в качестве следующего приближения берёт точку пересечения параболы и оси x.

Три изначально необходимых значения обозначаются как x_k, x_{k-1} и x_{k-2}. Парабола, проходящая через три точки $(x_k, f(x_k))$, $(x_{k-1}, f(x_{k-1}))$ и $(x_{k-2}, f(x_{k-2}))$ по формуле Ньютона записывается следующим образом

$$y = f(x_k) + (x - x_k)f[x_k, x_{k-1}] + (x - x_k)(x - x_{k-1})f[x_k, x_{k-1}, x_{k-2}],$$

где $f[x_k, x_{k-1}]$ и $f[x_k, x_{k-1}, x_{k-2}]$ суть разделённые разности.

Общая запись формулы:

$$x_{k+1} = x_k - \frac{2f(x_k)}{w \pm \sqrt{w^2 - 4f(x_k)f[x_k, x_{k-1}, x_{k-2}]}}.$$

где $w = f[x_k, x_{k-1}] + f[x_k, x_{k-2}] - f[x_{k-1}, x_{k-2}].$

Скорость сходимости метода Мюллера составляет примерно 1,84. Она лишь немного превышает порядок сходимости метода хорд (метода секущих). Это означает, что, несмотря на привлечение дополнительной информации о функции, метод парабол не увеличивает существенно порядок сходимости. Вместе с тем возникают задачи решения квадратного уравнения, выбора одного из двух корней многочлена и, самое важное, определения области гарантированной сходимости метода. Если три приближения для построения многочлена выбраны далеко от корня и содержат погрешности, то возможно самое неожиданное поведение решения.

Принцип остановки используется тот же из-за схожести методов.

2. Постановка задачи:

Для уравнений

(I) $x^2 - \sin(3x) - 1 = 0$ на интервале [−5, 5]

(II) $x^3 - 2x^2 + 1.2x - 0.226146 = 0$ на интервале [−2, 2]

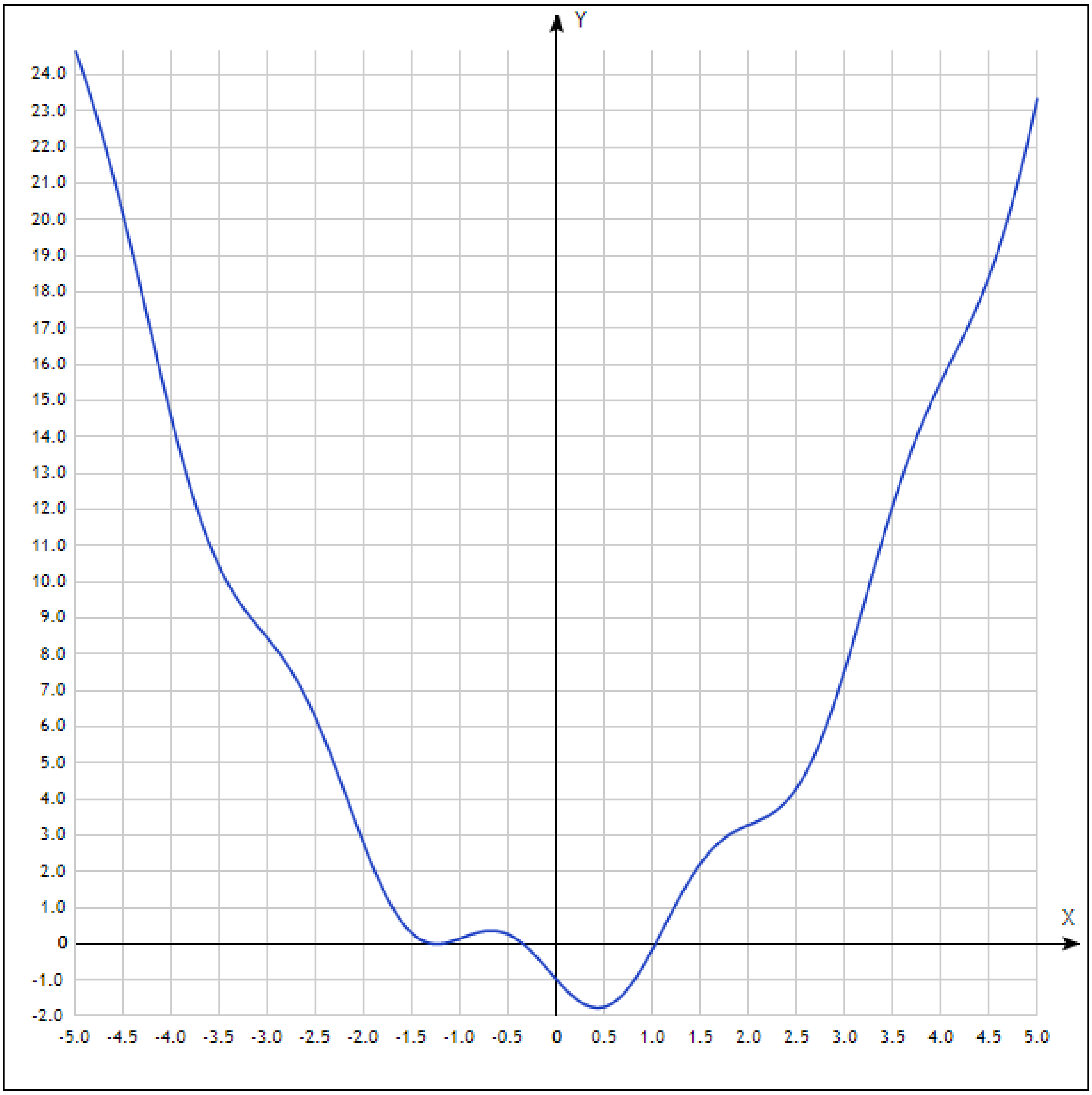
(III) $x^2 + x + 4\cos x - 4 = 0$ на интервале [−7, 7]

с помощью аналитического исследования и численных методов найти с точностью 10^{-10} все вещественные решения на заданных интервалах. Обоснование представить письменно и/или в виде результата корректно работающей программы.

В качестве численных методов применить метод секущих и метод Мюллера.

Сравнить реализованные методы по числу итераций и по общим трудозатратам.

(I) График функции:

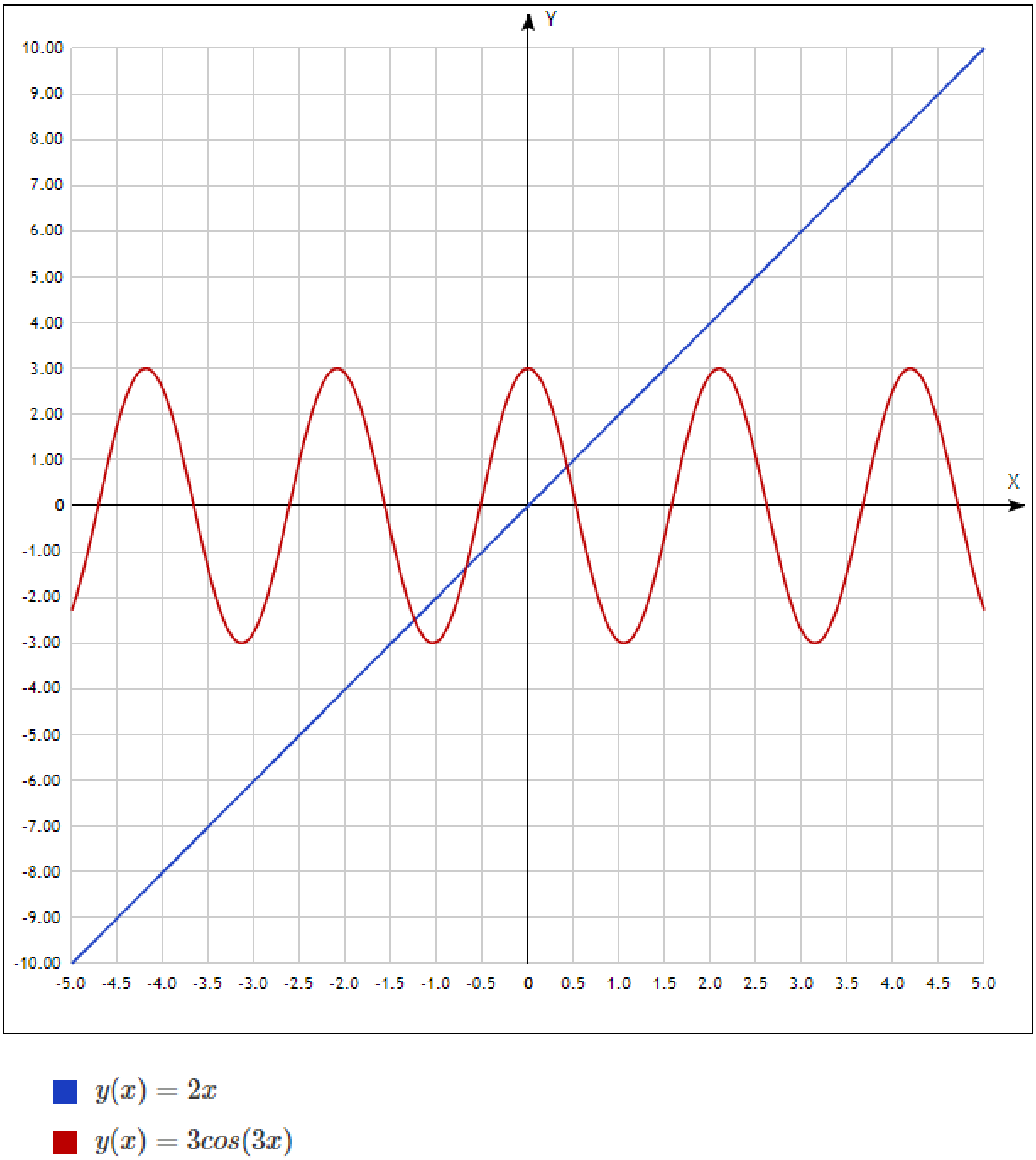


$y(x) = x^2 - sin(3x) - 1$

Найдем экстремумы функции и выясним ее знаки в этих точках, чтобы локализовать корни. Для их локализации мы можем использовать любые значения, лежащие рядом с двумя экстремумами или сами значения экстремумов, если функция в них разных знаков.

Найдем производную функции: $f_1'(x) = 2x - 3\cos(3x) = 0$

Строим графики $\begin{cases} y = 2x \\ y = 3\cos(3x) \end{cases}$:



Точки пересечения этих графиков – корни производной. У производной три корня, три экстремума у функции, значит, корней функции максимум четыре.

Возьмем приблизительные значения экстремумов, которые можно видеть из графика: -1.25, -0.7, 0.4. Точные знать необязательно, так как знак между двумя экстремумами меняется не более одного раза. В случае, если значения функций приблизительных значений соседних экстремумов одинакового знака, можно проверить другие приблизительные значения.

$f_1(-5) \approx 24.65 > 0$

$f_1(-1.25) \approx -0.01 < 0$

$f_1(-0.7) \approx 0.35 > 0$

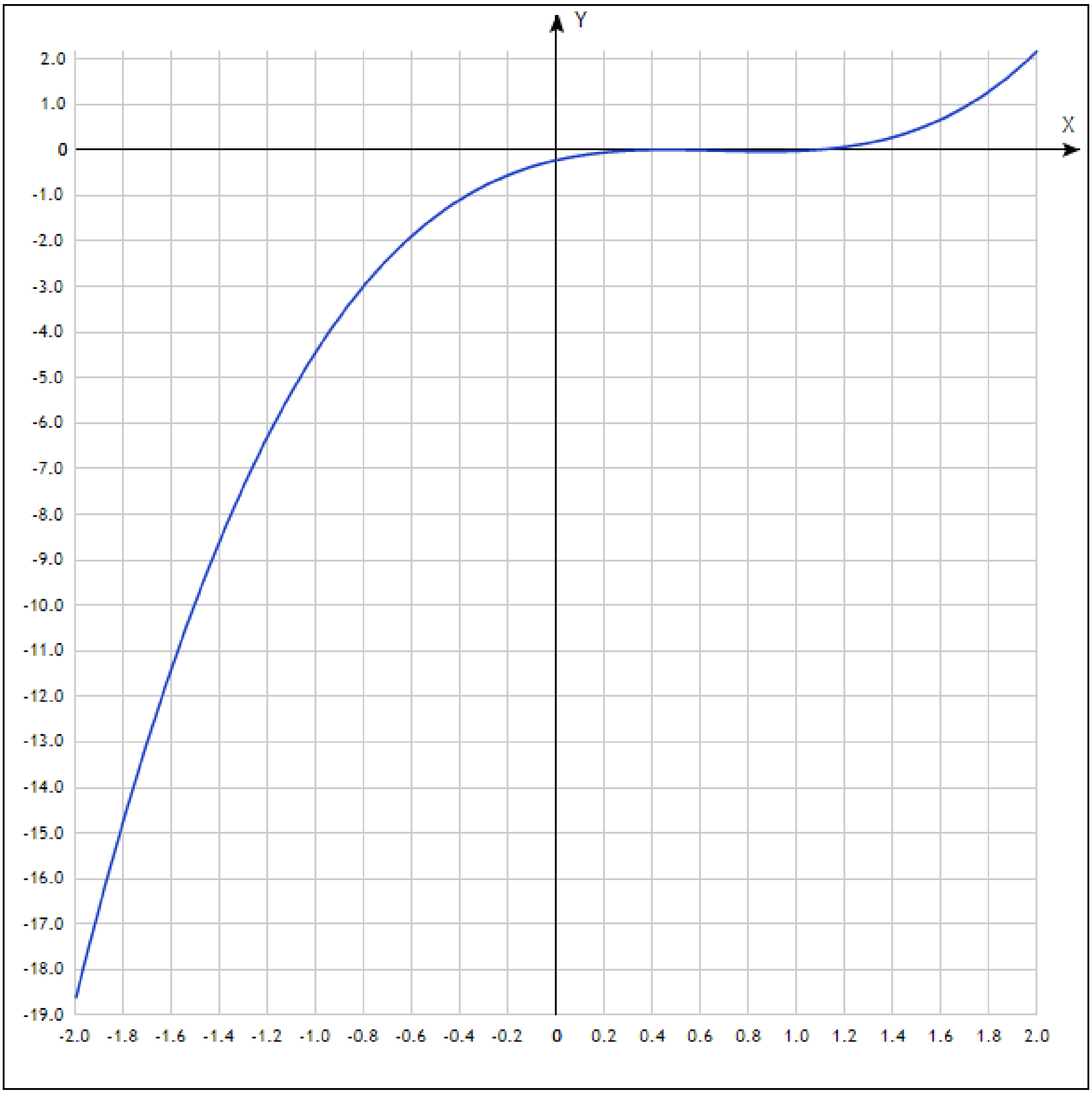
$f_1(0.4) \approx -1.77 < 0$

$f_1(5) \approx 23.4 > 0$

Знак меняется четыре раза, следовательно, корней тоже четыре.

Промежутки локализации корней: $x_1 \in [-5, -1.25], x_2 \in [-1.25, -0.7], x_3 \in [-0.7, 0.4]$, $x_4 \in [0.4, 5]$

(II) График функции:

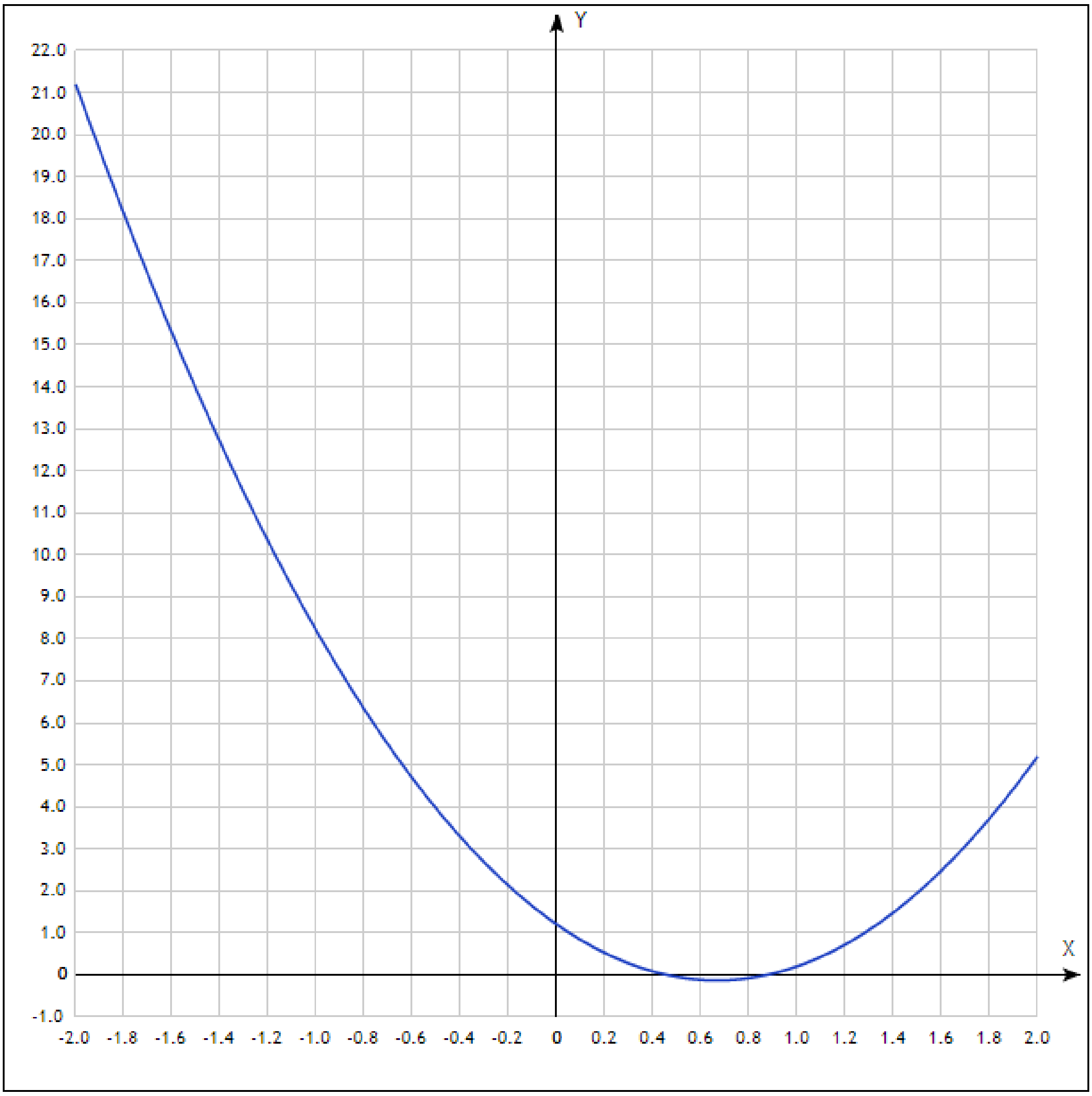


$y(x) = x^3 - 2x^2 + 1.2x - 0.226$

Найдем экстремумы функции.

Находим производную: $f_2'(x) = 3x^2 - 4x + 1.2 = 0$

Строим график этой производной функции:



$y(x) = 3x^2 - 4x + 1.2$

Найдем корни производных: $x_{1,2} = \frac{4 \pm 0.4\sqrt{10}}{6} = \frac{2 \pm 0.2\sqrt{10}}{3}$

Приблизительные экстремумы: 0.456 и 0.877. У производной два корня, два экстремума у функции, значит, корней функции максимум три.

$f_2(-2) \approx -18.63 < 0$

$f_2(0.456) \approx 0.0000008 > 0$

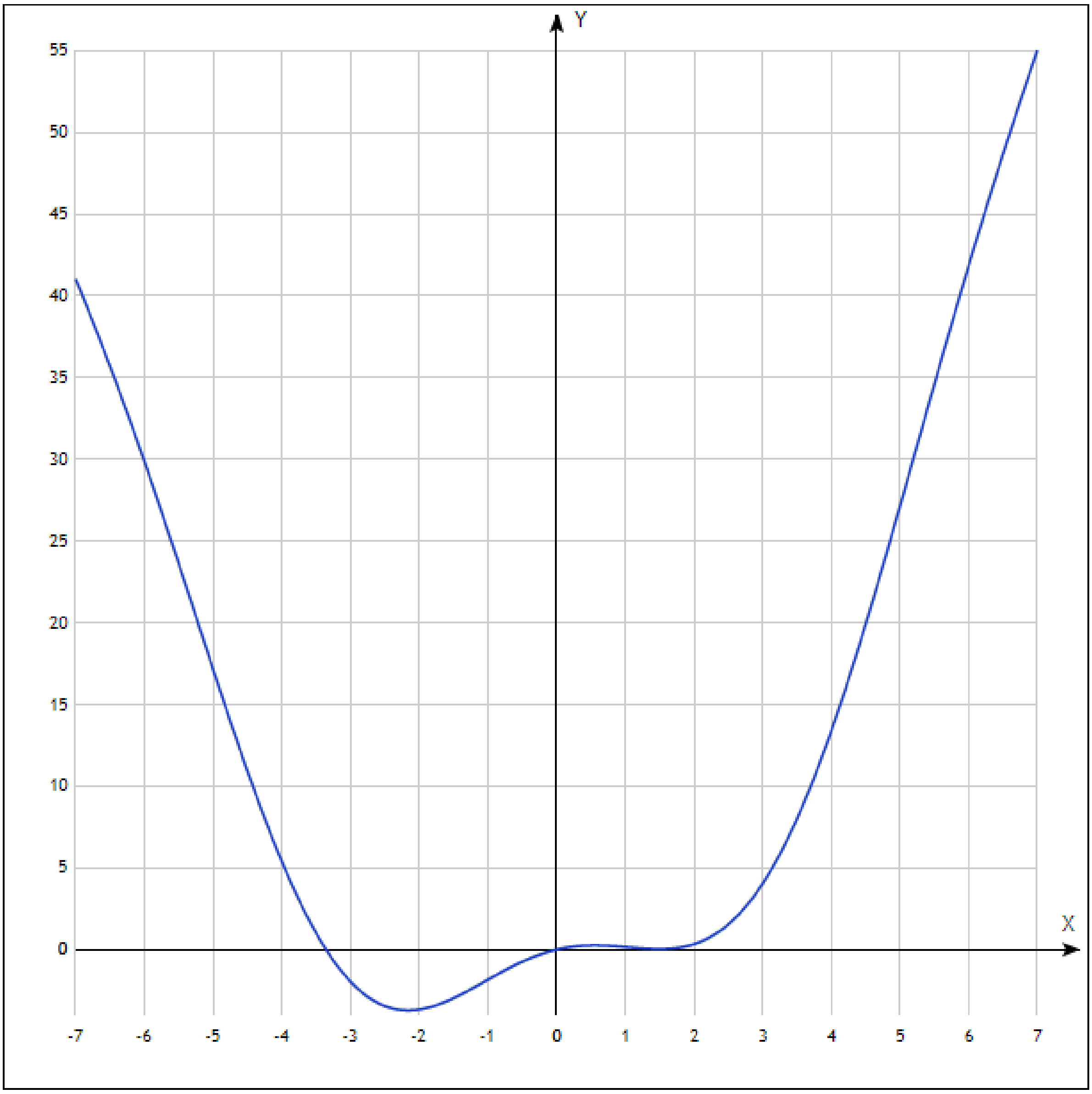
$f_2(0.877) \approx -0.04 < 0$

$f_2(2) \approx 2.17 > 0$

Знак меняется три раза, следовательно, корней тоже три.

Промежутки локализации корней: $x_1 \in [-2, 0.456], x_2 \in [0.456, 0.877], x_3 \in [0.877, 2]$

(III) График функции:

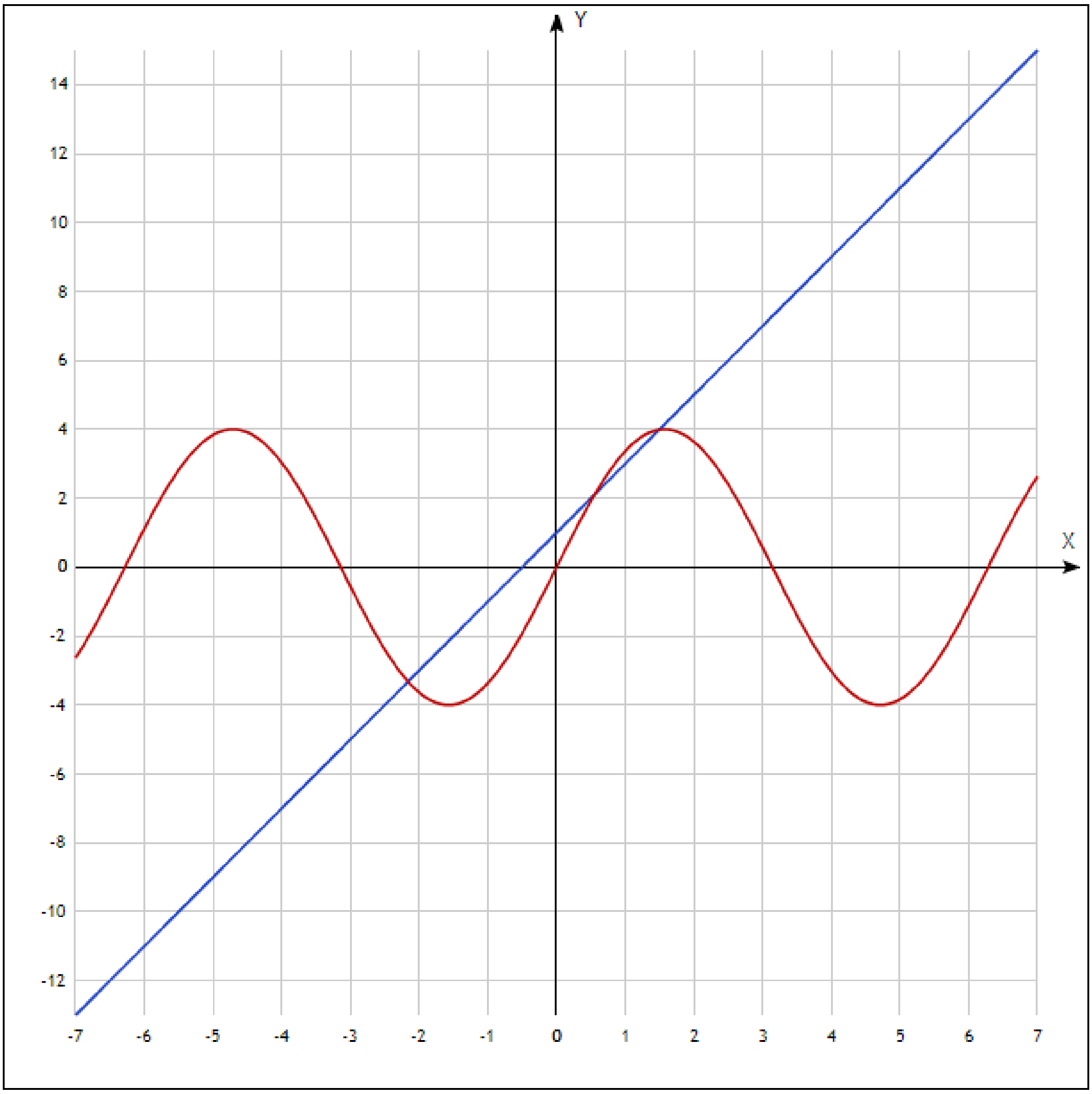


$y(x) = x^2 + x + 4cos(x) - 4$

Найдем экстремумы функции.

Находим производную: $f_3'(x) = 2x + 1 - 4\sin x = 0$

Строим графики $\begin{cases} y = 2x + 1 \\ y = 4\sin x \end{cases}$:



$y(x) = 2x + 1$

$y(x) = 4sin(x)$

У производной три корня, три экстремума у функции, значит, корней функции максимум четыре.

Приблизительные значения экстремумов: -2.2, 0.5, 1.5.

$f_3(-7) \approx 41.02 > 0$

$f_3(-2.2) \approx -3.71 < 0$

$f_3(0.5) \approx 0.26 > 0$

$f_3(1.5) \approx 0.03 > 0$

$f_3(7) \approx 55.02 > 0$

Известно, что функция между экстремумами монотонная. По графику производной видно, что значение первого корня точно меньше -2, а значение второго корня точно больше 0.1. Проверим значения функции в этих точках:

$f_3(-2) \approx -3.66 < 0$

$f_3(0.1) \approx 0.09 > 0$

Значит, на промежутке $[-2{,}0.1]$ есть корень, и он единственный, так как промежуток заведомо находится между двумя экстремумами функции.

На промежутке между следующими двумя экстремумами функция убывает. Проверим значение в точке 1:

$f_3(1) \approx 0.16 > 0$

Найдем промежуток, в котором лежит последний экстремум:

$f_3^I(1.45) \approx -0.07 < 0$

$f_3^I(1.5) \approx 0.01 > 0$

Значит, экстремум лежит в промежутке $[1.45{,}1.5]$.

$f_3(1.45) \approx 0.03 > 0$

Если корень у функции есть, то он лежит в этом интервале.

Построим уравнение касательной в точке 1.5. Значения этого уравнения будут меньше, чем значения функции в окрестности этого корня, так как мы рассматриваем точку на промежутке возрастания функции.

$y = f_3(1.5) + f_3^I(1.5)(x - 1.5)$

$y(1.45) \geq 0.03 + 0.01(1.45 - 1.5) > 0$

Касательная – функция линейная, и если бы на промежутке $[1.45{,}1.5]$ $f_3(x)$ принимала значение 0, то касательная бы принимала отрицательные значения в точках меньших экстремума. Поэтому можно сделать вывод, что корней функции на промежутке $[1.45{,}1.5]$ нет.

Следовательно, функция имеет только два корня.

Промежутки локализации корней: $x_1 \in [-7, -2.2], x_2 \in [-2.2, 0.5]$

3. Реализация и результаты:

При близких корнях методы могут выдавать неправильные корни (выходит за пределы промежутков первый корень второй функции):

```
C:\Windows\system32\cmd.exe
first function
Secant x1=-1.29441 iterations: 12
Parabol x1=-1.29441 iterations: 7
Secant x2=-1.19197 iterations: 10
Parabol x2=-1.19197 iterations: 6
Secant x3=-0.354624 iterations: 8
Parabol x3=-0.354624 iterations: 6
Secant x4=1.02815 iterations: 8
Parabol x4=1.02815 iterations: 8
second function
Secant x1=0.456995 iterations: 11
Parabol x1=-1.#IND iterations: 2
Secant x2=0.456995 iterations: 12
Parabol x2=0.456995 iterations: 5
Secant x3=1.0883 iterations: 14
Parabol x3=1.0883 iterations: 10
third function
Secant x1=-3.35626 iterations: 10
Parabol x1=-3.35626 iterations: 6
Secant x2=1.52649e-018 iterations: 10
Parabol x2=8.58745e-018 iterations: 7
Для продолжения нажмите любую клавишу . . .
```

Пришлось поменять промежуток этого корня: [0.45, 0.455].

Тогда корни уравнений такие:

```
C:\Windows\system32\cmd.exe
first function
Secant x1=-1.29441 iterations: 12
Parabol x1=-1.29441 iterations: 7
Secant x2=-1.19197 iterations: 10
Parabol x2=-1.19197 iterations: 6
Secant x3=-0.354624 iterations: 8
Parabol x3=-0.354624 iterations: 6
Secant x4=1.02815 iterations: 8
Parabol x4=1.02815 iterations: 8
second function
Secant x1=0.454703 iterations: 6
Parabol x1=0.454703 iterations: 4
Secant x2=0.456995 iterations: 12
Parabol x2=0.456995 iterations: 5
Secant x3=1.0883 iterations: 14
Parabol x3=1.0883 iterations: 10
third function
Secant x1=-3.35626 iterations: 10
Parabol x1=-3.35626 iterations: 6
Secant x2=1.52649e-018 iterations: 10
Parabol x2=8.58745e-018 iterations: 7
Для продолжения нажмите любую клавишу . . .
```

Вывод значений с 10 знаками после запятой (для проверки точности корней), а также со временем выполнения подсчета корня представлен в таблице:

Функция	Корень	Метод	Значение	Количество итераций	Время работы (наносекунды)
1	x_1	Секущие	-1.2944070481	12	17900
		Параболы	-1.2944070481	7	5000
	x_2	Секущие	-1.1919737206	10	4800
		Параболы	-1.1919737206	6	4200
	x_3	Секущие	-0.3546239851	8	4000
		Параболы	-0.3546239851	6	4100
	x_4	Секущие	1.0281537887	8	3800
		Параболы	1.0281537887	8	5300
2	x_1	Секущие	0.4547032161	6	2000
		Параболы	0.4547032161	4	2700
	x_2	Секущие	0.4569951723	12	3300
		Параболы	0.4569951723	5	2600
	x_3	Секущие	1.0883016116	14	3600
		Параболы	1.0883016116	10	4700
3	x_1	Секущие	-3.3562551339	10	5400
		Параболы	-3.3562551339	6	4100
	x_2	Секущие	0.0000000000	10	4500
		Параболы	0.0000000000	7	4600

При похожем количестве итераций у двух методов (иногда даже равном) и одинаковой точности корней при заданной погрешности подсчет в методе Мюллера сложнее и, соответственно, дольше. Нужно найти два корня и выбрать среди них один, тогда как метод секущих дает следующую итерацию в одно действие. Поэтому метод секущих выгоднее, чем метод Мюллера.

Наибольший выигрыш во времени метод Мюллера дает в первой функции. Для второй функции при данных промежутках он оказался невыгоден. Для третьей же функции он дает немного лучшие результаты, чем метод секущих.

Уменьшим промежутки (интервалы длиной 0.05, кроме второй функции, где один из корней вылетал из промежутка в методе секущих, а в методе парабол не выводил число):

Функция	Корень	Метод	Значение	Количество итераций	Время работы (наносекунды)
1	$[-1.3, -1.25]$	Секущие	-1.2944070481	7	16100
		Параболы	-1.2944070481	4	5000
	$[-1.2, -1.15]$	Секущие	-1.1919737206	6	4000
		Параболы	-1.1919737206	4	2700
	$[-0.4, -0.35]$	Секущие	-0.3546239851	5	2300
		Параболы	-0.3546239851	4	2100
	$[1, 1.05]$	Секущие	1.0281537887	4	2000
		Параболы	1.0281537887	6	3100
2	$[0.454, 0.455]$	Секущие	0.4547032161	6	1500
		Параболы	0.4547032161	4	1700
	$[0.456, 0.458]$	Секущие	0.4569951723	7	1700
		Параболы	0.4569951723	3	1300
	$[1.088, 1.09]$	Секущие	1.0883016116	4	1300
		Параболы	1.0883016116	3	1300
3	$[-3.4, -3.35]$	Секущие	-3.3562551339	4	2500
		Параболы	-3.3562551339	3	1700
	$[-0.03, 0.02]$	Секущие	0.0000000000	5	2400
		Параболы	0.0000000000	3	1800

Количество итераций обоих методов, как и время выполнения, уменьшились. Также мы видим, что метод Мюллера стал заметно более выгоден для третьей функции и немного лучше для второй функции, чем метод секущих.

Вывод:

Мы практически подтвердили, что скорость сходимости метода Мюллера выше (так как количество итераций практически во всех случаях меньше, чем у метода секущих), а также выяснили, что при уменьшении промежутков локализации корней метод Мюллера в основном тратит меньше времени на приближение к корню, чем метод секущих. Несмотря на более сложную структуру программы и выполнение большего количества операций за одну итерацию, он более выгоден в случае, если мы можем сузить промежутки. Если же по каким-то причинам этого сделать нельзя, то метод секущих на некоторых функциях может справиться лучше, чем метод Мюллера (что мы видели на примере второй функции).