НАО «Восточно-Казахстанский технический университет имени Д. Серикбаева»

Дисциплина: «Методы оптимизации»

ЛАБОРАТОРНАЯ РАБОТА № 1

ПОСТАНОВКА ЗАДАЧИ ОПТИМИЗАЦИИ. МЕТОДЫ ОДНОМЕРНОЙ ОПТИМИЗАЦИИ

Методические указания и задание к выполнению

Основа материала. Тема соответствует практической работе № 1 из силлабуса: «Постановка задачи оптимизации. Методы одномерной оптимизации». Теоретическая методичка, алгоритмы, программные шаблоны, варианты и форма отчёта добавлены для самостоятельного выполнения работы студентом.

1. Цель работы

Освоить постановку одномерной задачи оптимизации, научиться выбирать интервал поиска и критерий остановки, программно реализовывать методы дихотомии и золотого сечения, а также сравнивать их по точности и вычислительным затратам.

2. Планируемые результаты

- формулировать задачу минимизации функции одной переменной на заданном интервале;
- проверять пригодность интервала для поиска минимума и понимать условие унимодальности;
- реализовывать методы дихотомии и золотого сечения на Python;
- оценивать погрешность найденной точки и число вычислений целевой функции;
- представлять результаты в таблице и на графике и делать содержательный вывод.

3. Программное обеспечение

- Python 3.x;
- библиотеки NumPy и Matplotlib;
- любая среда разработки: Jupyter Notebook, VS Code, PyCharm или аналогичная.

4. Методические сведения перед выполнением

4.1. Постановка одномерной задачи

Одномерная оптимизация рассматривает функцию f(x), зависящую от одной переменной. Требуется найти точку x* на интервале [a,b], в которой достигается минимальное значение:

$$f(x^*) = \min f(x), \quad x \in [a,b].$$

Для задачи максимизации можно минимизировать функцию −f(x). Численные методы обычно требуют, чтобы на выбранном интервале функция была унимодальной: сначала убывала, затем возрастала, имея один минимум.

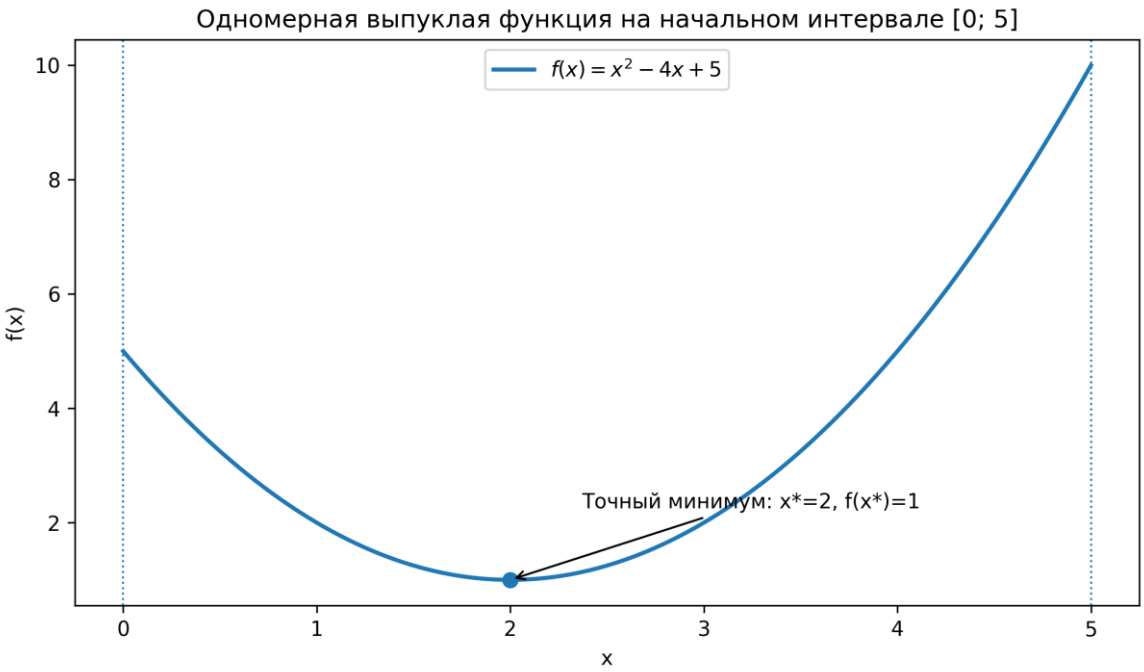


Рисунок 1 — Пример одномерной задачи минимизации

4.1.1. Глобальный и локальный минимум

Точка $x^* \in D$ называется точкой глобального минимума функции f, если $f(x^*) \le f(x)$ для всех $x \in D$. Точка x^* называется точкой локального минимума, если существует $\delta > 0$ такое, что $f(x^*) \le f(x)$ для всех $x \in D \cap (x^* - \delta, x^* + \delta)$. Глобальный минимум автоматически является локальным, но обратное в общем случае неверно.

$$x^* \text{ — глобальный минимум} \Leftrightarrow f(x^*) \le f(x) \text{ для всех } x \in D.$$

$$x^* \text{ — локальный минимум} \Leftrightarrow \exists \delta > 0 : f(x^*) \le f(x) \text{ при } |x - x^*| < \delta.$$

Для численных методов важно заранее понимать, ищется ли глобальный минимум на всём допустимом множестве или локальный минимум в заданной окрестности.

4.1.2. Монотонность и унимодальность

Функция монотонно возрастает на интервале, если $x_1 < x_2$ влечёт $f(x_1) \le f(x_2)$, и монотонно убывает, если $x_1 < x_2$ влечёт $f(x_1) \ge f(x_2)$. Для методов исключения интервалов ключевое предположение — унимодальность.

Функция f называется унимодальной на [a,b], если существует единственная точка $x^* \in [a,b]$, такая что f убывает слева от x^* и возрастает справа от x^*. Непрерывность удобна, но само понятие унимодальности прежде всего описывает форму функции и единственность минимума на интервале.

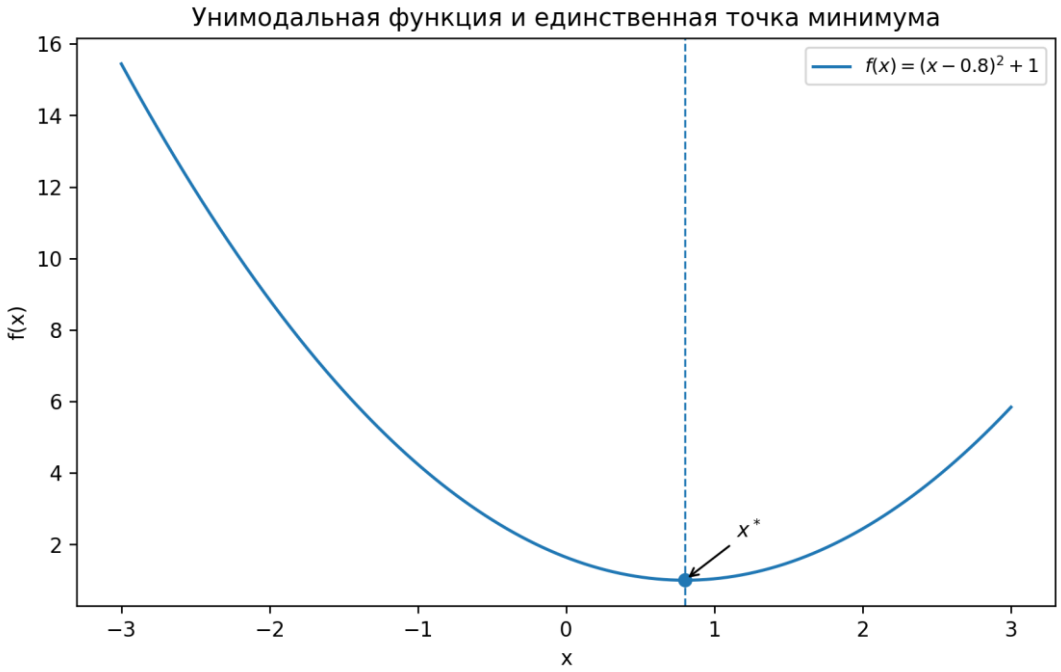


*Рисунок 1а — Унимодальная функция: убывание слева и возрастание справа от x**

4.1.3. Аналитическая проверка экстремума

Если функция дифференцируема, внутреннюю точку минимума обычно ищут из необходимого условия $f'(x^*)=0$. Для дважды дифференцируемой функции достаточным локальным условием является $f'(x^*)=0$ и $f''(x^*)>0$. После нахождения стационарных точек необходимо сравнить значения функции также на границах [a,b].

$$f'(x^*)=0, \quad f''(x^*)>0 \Rightarrow x^* \text{ — строгий локальный минимум.}$$

В лабораторной работе аналитическое решение используется как контрольное. Численный алгоритм должен находить близкий результат, не используя точное значение x* внутри самого метода.

4.1.4. Прямой поиск: пассивная и последовательная стратегии

Если производные недоступны или их вычисление ненадёжно, используют прямой поиск, основанный только на значениях f(x). В пассивной стратегии все точки выбираются заранее; в последовательной следующая точка зависит от ранее вычисленных значений функции.

Для оптимального пассивного поиска N точек можно взять равномерно:

$$x_k = a + k(b-a)/(N+1), \quad k=1,\ldots,N.$$

Если минимальное из вычисленных значений получено в x_j, то при унимодальности минимум локализуется между соседними точками. Внутренний интервал неопределённости имеет длину приблизительно

$$l_N = 2(b-a)/(N+1).$$

Последовательные методы — дихотомия, золотое сечение, Фибоначчи — используют уже накопленную информацию и обычно достигают той же точности при меньшем числе вычислений функции.

4.2. Критерий остановки

Последовательные методы уменьшают интервал неопределённости. Вычисления прекращаются, когда длина интервала становится достаточно малой:

$$b_k - a_k \le \varepsilon,$$

где $\varepsilon > 0$ — заданная точность. В качестве приближённого решения обычно принимают середину последнего интервала $x^* \approx (a_k+b_k)/2$.

4.3. Метод дихотомии

На каждой итерации около середины интервала выбирают две близкие точки x_1 и x_2. Значения функции сравниваются, после чего часть интервала, которая не может содержать минимум унимодальной функции, отбрасывается.

$$x_1 = (a+b)/2 - \delta, \quad x_2 = (a+b)/2 + \delta, \quad 0 < \delta < \varepsilon/2.$$

1. Задать [a,b], точность ε и малое число δ.
2. Вычислить x_1, x_2 и значения $f(x_1)$, $f(x_2)$.
3. Если $f(x_1) \le f(x_2)$, принять $b := x_2$; иначе $a := x_1$.
4. Повторять, пока $b-a > \varepsilon$.
5. Принять $x^* = (a+b)/2$.

4.3.1. Теоретическая оценка сходимости дихотомии

Пусть $l_k = b_k - a_k$ — длина текущего интервала. После одной итерации дихотомии сохраняется интервал длины

$$l_{k+1} = l_k/2 + \delta.$$

Решая это рекуррентное соотношение, получаем

$$l_k = 2\delta + (l_0 - 2\delta)/2^k.$$

Поэтому требование $l_k \le \varepsilon$ возможно только при $\varepsilon > 2\delta$. Минимальное число сокращений можно оценить как

$$k \ge \log_2((l_0 - 2\delta)/(\varepsilon - 2\delta)).$$

Именно поэтому δ нельзя выбирать слишком большим. В программной реализации удобно, например, брать $\delta = \varepsilon/4$.

4.4. Метод золотого сечения

Метод использует постоянное отношение частей интервала, благодаря чему после первой итерации одно из ранее вычисленных значений функции можно использовать повторно. Это уменьшает число новых вычислений f(x).

$$\tau = (\sqrt{5} - 1)/2 \approx 0{,}618034$$

$$x_1 = b - \tau(b-a), \quad x_2 = a + \tau(b-a).$$

1. Вычислить внутренние точки x_1, x_2 и значения $f(x_1)$, $f(x_2)$.
2. Если $f(x_1) \le f(x_2)$, отбросить правую часть интервала; иначе — левую.
3. Перенести одну внутреннюю точку и её значение функции в новый интервал.
4. Вычислить только одну новую точку.
5. Продолжать до выполнения $b-a \le \varepsilon$.

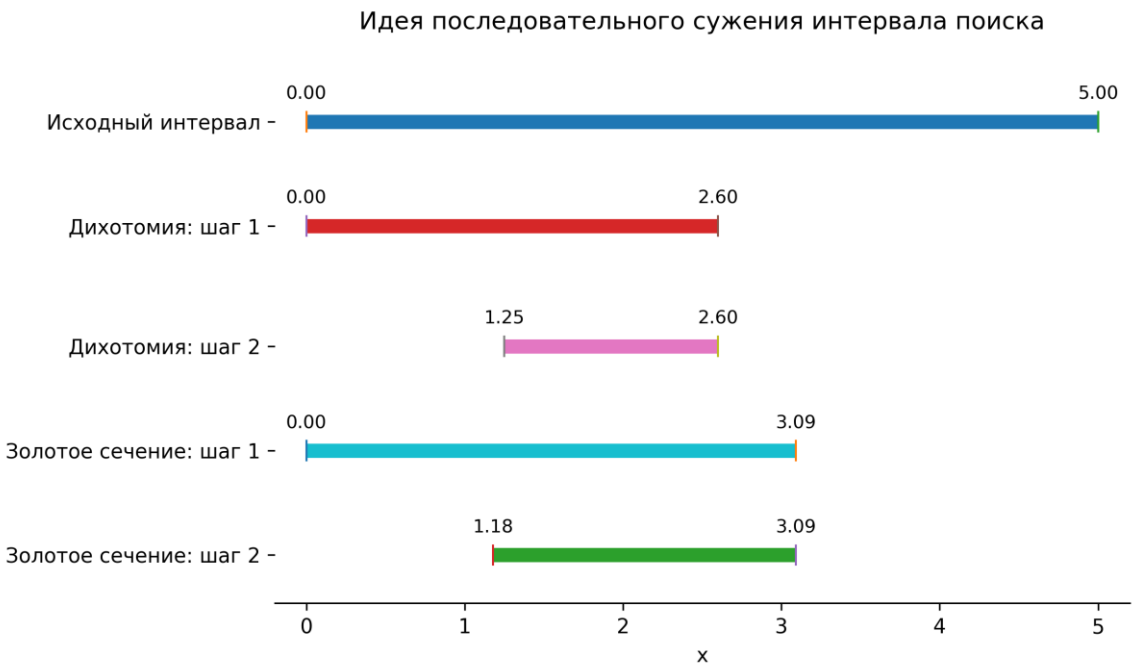


Рисунок 2 — Сужение интервала в последовательных методах поиска

4.4.1. Вывод коэффициента золотого сечения

Преимущество золотого сечения возникает из требования: после отбрасывания части интервала одна из старых внутренних точек должна стать внутренней точкой нового интервала. Если отношение новой длины к старой постоянно и равно τ, то самоподобие интервалов приводит к уравнению

$$\tau^2 + \tau - 1 = 0.$$

Положительный корень равен

$$\tau = (\sqrt{5}-1)/2 \approx 0{,}618034.$$

После k сокращений длина интервала равна

$$l_k = \tau^k l_0.$$

Следовательно, для $l_k \le \varepsilon$ достаточно

$$k \geq \ln(l_0/\varepsilon) \,/\, \ln(1/\tau).$$

После первых двух вычислений f метод золотого сечения требует только одного нового вычисления функции на каждое сокращение интервала. Это позволяет сравнивать методы не только по числу итераций, но и по реальной вычислительной стоимости.

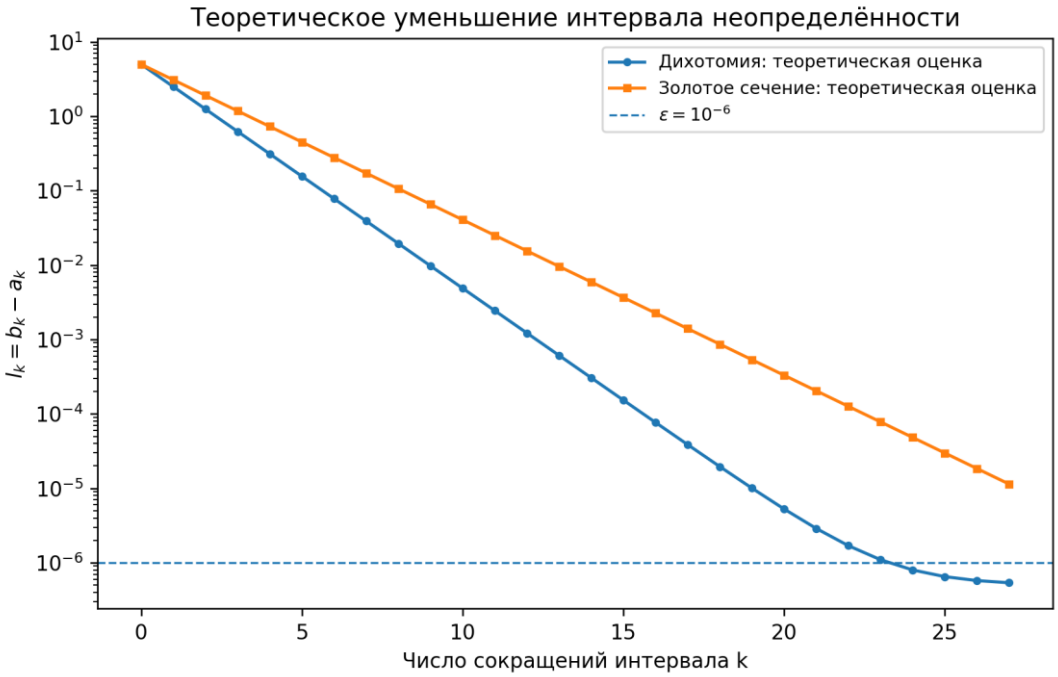


Рисунок 2а — Теоретические оценки уменьшения интервала неопределённости

4.5. Что сравнивать в вычислительном эксперименте

Для корректного сравнения оба метода должны решать одну и ту же задачу при одинаковом начальном интервале и одинаковой точности ε. Рекомендуется фиксировать: приближённую точку x*, значение f(x*), конечную длину интервала, число итераций, число вычислений функции и абсолютную ошибку относительно аналитического решения, если оно известно.

4.6. Карта методов одномерной оптимизации

Класс	Методы	Используемая информация	Комментарий
Прямой пассивный поиск	равномерный поиск	только f(x)	прост, но заранее фиксирует все точки
Прямой последовательный поиск	дихотомия, деление пополам, золотое сечение, Фибоначчи	только f(x)	надёжное сужение интервала при унимодальности
Полиномиальная аппроксимация	Пауэлл	f(x) в нескольких точках	эффективен для квадратичной или близкой к квадратичной функции
Производные	средняя точка, Ньютон, секущие	f′(x), иногда f″(x)	быстрее локально, но требуют гладкости и хорошего начального приближения
Кубическая аппроксимация	кубический интерполяционный поиск	f(x) и f′(x)	высокая точность для гладких функций

В этой лабораторной подробно реализуются дихотомия и золотое сечение. Остальные методы приведены для математической классификации и будут последовательно развиваться в следующих практических работах.

5. Разобранный пример

Рассмотрим функцию $f(x)=x^2-4x+5$ на интервале $[0;5]$. Аналитически $f'(x)=2x-4$, поэтому точный минимум достигается при $x=2$, а $f(2)=1$. Этот результат используется только для проверки численных методов.

$$f(x)=x^2-4x+5, \quad [a,b]=[0,5], \quad \varepsilon=10^{-4}.$$

После выполнения численного алгоритма найденная точка должна быть близка к 2. Разность $|x_num-2|$ характеризует ошибку по аргументу.

5.1. Первые итерации метода золотого сечения вручную

Для функции $f(x)=x^2-4x+5$ на [0;5] коэффициент $\tau\approx0{,}618034$. На первой итерации $x_1=5-\tau\cdot5\approx1{,}90983$, $x_2=\tau\cdot5\approx3{,}09017$. Поскольку $f(x_1)<f(x_2)$, правая часть интервала отбрасывается и новым интервалом становится [0;3,09017]. Одна из внутренних точек переносится на следующую итерацию без повторного вычисления функции.

k	a_k	b_k	x_1	x_2	$f(x_1)$	$f(x_2)$	b_k-a_k
1	0.00000	3.09017	1.18034	1.90983	1.671843	1.008131	3.09017
2	1.18034	3.09017	1.90983	2.36068	1.008131	1.130090	1.90983
3	1.18034	2.36068	1.63119	1.90983	1.136021	1.008131	1.18034
4	1.63119	2.36068	1.90983	2.08204	1.008131	1.006730	0.72949
5	1.90983	2.36068	2.08204	2.18847	1.006730	1.035521	0.45085

Таблица показывает главное свойство последовательного поиска: интервал, содержащий минимум, монотонно сужается. При этом золотое сечение после начальной пары значений функции требует только одного нового вычисления f на каждой итерации.

5.2. Контрольный вычислительный эксперимент

Для точного решения $x_{точн}=2$ сравним оба метода при трёх значениях ε. Число вычислений f включает итоговое вычисление $f(x^*)$.

Метод	ε	x^*	$f(x^*)$	Итерации	Вычисления f	$\lvert x^*-2\rvert$
Дихотомия	1e-02	2.00001221	1.0000000001	10	21	1.22e-05
Золотое сечение	1e-02	2.00186590	1.0000034816	13	16	1.87e-03
Дихотомия	1e-04	1.99999356	1.0000000000	17	35	6.44e-06
Золотое сечение	1e-04	1.99998483	1.0000000002	23	26	1.52e-05
Дихотомия	1e-06	2.00000008	1.0000000000	24	49	7.98e-08
Золотое сечение	1e-06	2.00000012	1.0000000000	33	36	1.23e-07

По таблице видно, что оба метода сходятся к x=2. Дихотомия обычно делает меньше итераций, но на каждой итерации заново вычисляет функцию в двух точках. Золотое сечение использует повторно одно значение и поэтому требует меньше вычислений целевой функции, что особенно важно, если один вызов f(x) вычислительно дорог.

6. Шаблон программы на Python

Студент должен разобраться в логике функций и при необходимости дополнить их счётчиком вычислений целевой функции и сохранением истории интервалов.

```
import math


def f(x):
    return x**2 - 4*x + 5


def dichotomy(f, a, b, eps=1e-4):
    delta = eps / 4
    it = 0
    while b - a > eps:
        x1 = (a + b) / 2 - delta
        x2 = (a + b) / 2 + delta
        if f(x1) <= f(x2):
            b = x2
        else:
            a = x1
        it += 1
    x = (a + b) / 2
```

```
    return x, f(x), it


def golden_section(f, a, b, eps=1e-4):
    tau = (math.sqrt(5) - 1) / 2
    x1 = b - tau * (b - a)
    x2 = a + tau * (b - a)
    f1, f2 = f(x1), f(x2)
    it = 0
    while b - a > eps:
        if f1 <= f2:
            b, x2, f2 = x2, x1, f1
            x1 = b - tau * (b - a)
            f1 = f(x1)
        else:
            a, x1, f1 = x1, x2, f2
            x2 = a + tau * (b - a)
            f2 = f(x2)
        it += 1
    x = (a + b) / 2
    return x, f(x), it

print('Дихотомия:', dichotomy(f, 0, 5))
print('Золотое сечение:', golden_section(f, 0, 5))
```

6.1. Рекомендуемая полная реализация с историей итераций

```
import math
import numpy as np
import matplotlib.pyplot as plt

def f(x):
    return x**2 - 4*x + 5

def dichotomy(f, a, b, eps=1e-4):
    delta = eps / 4
    history = []
    nfev = 0
    k = 0
    while b - a > eps:
        k += 1
        x1 = (a + b)/2 - delta
        x2 = (a + b)/2 + delta
        f1, f2 = f(x1), f(x2)
        nfev += 2
        if f1 <= f2:
            b = x2
        else:
            a = x1
        history.append((k, a, b, b-a))
    x_star = (a + b)/2
    return x_star, f(x_star), k, nfev + 1, history

def golden_section(f, a, b, eps=1e-4):
    tau = (math.sqrt(5) - 1)/2
    x1 = b - tau*(b-a)
    x2 = a + tau*(b-a)
    f1, f2 = f(x1), f(x2)
    nfev = 2
    k = 0
    history = []
    while b - a > eps:
        k += 1
        if f1 <= f2:
            b, x2, f2 = x2, x1, f1
            x1 = b - tau*(b-a)
```

```
            f1 = f(x1)
        else:
            a, x1, f1 = x1, x2, f2
            x2 = a + tau*(b-a)
            f2 = f(x2)
        nfev += 1
        history.append((k, a, b, b-a))
    x_star = (a + b)/2
    return x_star, f(x_star), k, nfev + 1, history

for eps in [1e-2, 1e-4, 1e-6]:
    print('eps =', eps)
    print('dichotomy:', dichotomy(f, 0, 5, eps)[:4])
    print('golden   :', golden_section(f, 0, 5, eps)[:4])
```

6.2. Графическая проверка решения и сходимости

Помимо графика самой функции, полезно строить зависимость длины интервала $b_k - a_k$ от номера итерации. Логарифмический масштаб по вертикальной оси делает геометрическое уменьшение интервала особенно наглядным.

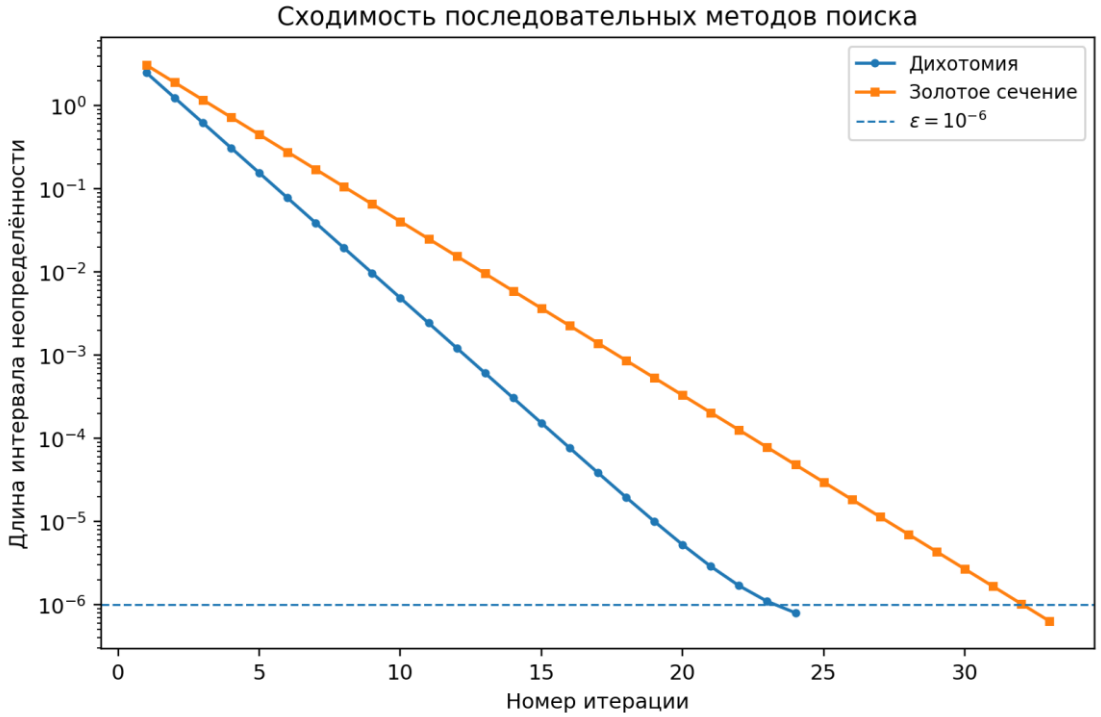


Рисунок 3 — Сравнение сужения интервала методом дихотомии и методом золотого сечения при $\varepsilon=10^{-6}$

Для отчёта достаточно одного графика функции с отмеченной точкой минимума и одного графика сходимости. По второму графику студент должен объяснить, какой метод быстрее уменьшает интервал по числу итераций и какой экономичнее по числу вычислений f.

7. Порядок выполнения лабораторной работы

1. Получить у преподавателя номер варианта.
2. Записать свою функцию f(x), интервал [a,b] и точность ε.
3. Построить график функции на заданном интервале и визуально убедиться в наличии одного минимума.
4. По возможности найти аналитическое решение через условие $f'(x)=0$ и проверить, что оно принадлежит интервалу.
5. Реализовать метод дихотомии. Добавить счётчик итераций и вычислений функции.

6. Реализовать метод золотого сечения с повторным использованием одного из значений функции.
7. Выполнить вычисления для $\varepsilon = 10^{-2}$, 10^{-4} и 10^{-6}.
8. Свести результаты в сравнительную таблицу.
9. Построить итоговый график функции, отметить найденный минимум.
10. Сформулировать вывод о точности и вычислительной эффективности методов.

8. Варианты индивидуальных заданий

Вариант	Целевая функция	Интервал
1	$f(x)=(x-1.5)^2+2$	[−1; 4]
2	$f(x)=2x^2-8x+10$	[0; 5]
3	$f(x)=3x^2+6x+7$	[−4; 2]
4	$f(x)=0.5x^2-3x+6$	[0; 8]
5	$f(x)=4x^2-4x+3$	[−1; 3]
6	$f(x)=x^2+2x+5$	[−5; 2]
7	$f(x)=1.5x^2-9x+20$	[0; 8]
8	$f(x)=2.5x^2+5x+4$	[−4; 2]
9	$f(x)=0.8x^2-4x+7$	[0; 6]
10	$f(x)=5x^2-20x+25$	[0; 5]
11	$f(x)=1.2x^2+4.8x+8$	[−5; 1]
12	$f(x)=3.5x^2-7x+6$	[−1; 4]
13	f(x)=(x−2)^4+0.5(x−2)^2+1	[−1; 5]
14	f(x)=exp(0.2x)+(x−1)^2	[−2; 4]
15	$f(x)=(x+1)^2+\sin^2(x)$	[−3; 1]
16	$f(x)=x^2+\exp(-x)$	[−2; 3]
17	f(x)=(x−1)^2+0.2 cos(x)	[−2; 4]
18	$f(x)=x^2+\ln(1+x^2)$	[−3; 3]

Примечание к вариантам 13–18. Для нелинейных функций точное решение не обязательно выражается простой формулой. Аналитическая проверка может заключаться в исследовании производной и подтверждении единственности минимума; для контрольного значения допускается использовать высокоточную численную проверку с существенно меньшим ε.

8.1. Дополнительный набор тестовых функций из методических указаний

Для углублённого варианта работы можно использовать тестовые функции из методических указаний Т. М. Поповой. Они подобраны так, чтобы охватить тригонометрические, полиномиальные и экспоненциальные функции.

№	f(x)	Интервал
1	sin x	[−π; π/2]
2	cos x	[0; π]
3	$(x-2)^2$	[0; 3]
4	$(x-15)^2+5$	[12; 20]
5	$(x+5)^4$	[−6; 2]
6	$x\cdot e^x$	[−2; 0]
7	x^2+2x-4	[−2; 1]
8	x^3-x	[0; 1]
9	x^5-x^2	[0; 1]
10	$-x/e^x$	[0; 3]
11	x^4-x	[0; 1]
12	$x^4/\ln x$	[1,1; 1,5]
13	x·e^(−x)	[−2; 6]
14	x·e^(−2x)	[−2; 6]

В расширенном режиме студент может дополнительно сравнить один из интервальных методов с методом, использующим производные или аппроксимацию. В исходных методических указаниях предлагается: для вариантов 1–4 — пассивный поиск, золотое сечение и Ньютон; 5–8 — дихотомия,

Фибоначчи и секущие; 9–12 — деление пополам, Пауэлл и Ньютон; 13–14 — золотое сечение, Пауэлл и кубическая аппроксимация.

9. Таблица для результатов

Метод	ε	x*	f(x*)	Итерации	Вычисления f	|x*−xточн|	b−a

9.0. Расширенная таблица сходимости

Для каждой итерации рекомендуется сохранять не только границы a_k и b_k, но и длину интервала $l_k=b_k-a_k$ и коэффициент фактического сжатия $q_k=l_{k-1}/l_k$. Для золотого сечения после переходного шага q_k стремится к $1/\tau\approx1{,}618$. Для дихотомии при малом δ коэффициент близок к 2.

k	a_k	b_k	$l_k=b_k-a_k$	l_{k-1}/l_k	x_1	x_2	$f(x_1)$	$f(x_2)$
…	…	…	…	…	…	…	…	…

Дополнительно постройте график зависимости числа вычислений целевой функции N_f от $\log_{10}(1/\varepsilon)$ для $\varepsilon=10^{-2},10^{-3},\ldots,10^{-7}$. Такой график наглядно показывает, как вычислительная стоимость растёт при ужесточении требования к точности.

9.1. Чек-лист перед оформлением отчёта

- на одном и том же варианте выполнены оба метода;
- для каждого ε зафиксированы x*, f(x*), число итераций и число вычислений функции;
- программа действительно повторно использует одно значение f в методе золотого сечения;
- результат проверен аналитически или независимым высокоточным вычислением;
- графики имеют подписи осей, заголовок и отметку найденного минимума;
- в выводе сравниваются не только значения x*, но и вычислительная стоимость методов.

10. Требования к отчёту

- титульная часть: дисциплина, номер и тема работы, Ф.И.О. студента, группа;
- цель работы;
- индивидуальное задание и математическая постановка;
- краткое описание методов дихотомии и золотого сечения;
- листинг программы;
- график целевой функции;
- таблица вычислительных результатов;
- сравнительный анализ методов;
- вывод;
- ответы на контрольные вопросы.

11. Критерии оценивания

Элемент работы	Баллы
Постановка задачи и аналитическая проверка	15

Корректная реализация дихотомии	20
Корректная реализация золотого сечения	20
Вычислительный эксперимент для трёх ε	15
Графики и таблица результатов	10
Анализ и вывод	10
Защита и ответы на вопросы	10

12. Контрольные вопросы

1. Что означает унимодальность функции на интервале?
2. Как выбирается критерий остановки последовательного поиска?
3. Какова роль параметра δ в методе дихотомии?
4. Почему в методе золотого сечения на новой итерации требуется только одно новое вычисление функции?
5. Как влияет уменьшение ε на число итераций?
6. Можно ли применять эти методы непосредственно к функции с несколькими локальными минимумами на одном интервале?
7. Как перейти от задачи максимизации к задаче минимизации?
8. Какие показатели следует использовать для сравнения численных методов оптимизации?
9. Чем глобальный минимум отличается от локального?
10. Докажите формулу $l_k = 2\delta + (l_0 - 2\delta)/2^k$ для дихотомии.
11. Из какого уравнения получается коэффициент золотого сечения τ?
12. Почему число вычислений функции — более информативная характеристика, чем только число итераций?
13. В каких случаях метод Ньютона может быть предпочтительнее интервального поиска, а в каких — менее надёжен?

13. Пример формулировки вывода

Образец. В работе реализованы методы дихотомии и золотого сечения для одномерной унимодальной функции. При уменьшении заданной погрешности найденная точка приблизилась к аналитическому минимуму, а число итераций увеличилось. Метод золотого сечения оказался экономичнее по числу новых вычислений целевой функции благодаря повторному использованию одного из значений на каждой следующей итерации. Конкретные численные показатели в отчёте должны быть приведены по своему варианту.

Литература по силлабусу

1. Васильев Ф. П. и др. Методы оптимизации: учебник и практикум для бакалавриата и магистратуры. — М.: Юрайт, 2019. — 375 с.
2. Струченков В. И. Методы оптимизации в прикладных задачах. — М.: СОЛОН-ПРЕСС, 2012. — 320 с.
3. Казанская О. В., Юн С. Г., Альсова О. К. Модели и методы оптимизации. Практикум. — Новосибирск: НГТУ, 2012. — 204 с.
4. Черняк А. А., Черняк Ж. А., Метельский Ю. М., Богданович С. А. Методы оптимизации: теория и алгоритмы. — 2-е изд. — М.: Юрайт, 2023. — 357 с.
5. Седых И. А., Истомин В. А., Сёмина В. В., Супрунов И. И. Методы оптимизации: теория, практические задания и лабораторные работы. — 2025.

6. Попова Т. М. Методы одномерной оптимизации: методические указания и задания к выполнению лабораторных работ по дисциплине «Методы оптимизации». — Хабаровск: Изд-во Тихоокеанского государственного университета, 2011. — 26 с.

7. Аттетков А. В., Галкин С. В., Зарубин В. С. Методы оптимизации. — 2-е изд. — М.: Изд-во МГТУ им. Н. Э. Баумана, 2003. — 440 с.

8. Алексеев В. М., Галлеев Э. М., Тихомиров В. М. Сборник задач по оптимизации. — М.: Наука, 1984. — 287 с.

9. Ашманов С. А., Тимохов А. В. Теория оптимизации в задачах и упражнениях. — М.: Наука, 1991. — 448 с.