



Министерство науки и высшего образования Российской Федерации
Федеральное государственное бюджетное образовательное учреждение
высшего образования
«Московский государственный технический университет
имени Н.Э. Баумана
(национальный исследовательский университет)»
(МГТУ им. Н.Э. Баумана)

ФАКУЛЬТЕТ *Робототехники и комплексной автоматизации*

КАФЕДРА *Системы автоматизированного проектирования (РК-6)*

ОТЧЕТ О ВЫПОЛНЕНИИ ЛАБОРАТОРНОЙ РАБОТЫ

по дисциплине: «Вычислительная математика»

Студент

Шпиталев Даниил Олегович

Группа

РК6-56Б

Тип задания

лабораторная работа 2 Вариант 5

Тема лабораторной работы

Использование аппроксимаций для
численной оптимизации

Студент

подпись, дата

Шпиталев Д.О.

фамилия, и.о.

Преподаватель

подпись, дата

Соколов А.П.

фамилия, и.о.

Москва, 2023 г.

Оглавление

Задание на лабораторную работу	3
Цель выполнения лабораторной работы	5
1. Базовая часть.....	5
1.1. Разработка функций численного интегрирования с помощью составной формулы Симпсона и составной формулы трапеций.....	5
1.2. Расчёт интеграла (1) для функции $y(x)$, соответствующей кривой наискорейшего спуска, с помощью составной формулы Симпсона и составной формулы трапеций.	7
1.3. Построение графика зависимости абсолютных погрешностей численного интегрирования в зависимости от шага интегрирования.	9
1.4. Определение порядка точности формул по графику.	10
1.5. Сравнение известных аналитических порядков точности с порядками точности, получаемыми с помощью соответствующих графиков.....	10
1.6. Существование оптимального шага интегрирования для первой и второй формулы, минимизирующего достижимую погрешность.	12
2. Продвинутая часть	12
2.1. Преобразование задачи о минимизации функционала к полудискретной форме с помощью кусочно-линейной интерполяции.	12
2.2. Преобразование функционала к полностью дискретной форме с помощью составной формулы Симпсона.....	13
Заключение	15
Список использованных источников	16

Задание на лабораторную работу

Методы аппроксимации, такие как интерполяция и численное интегрирование, часто используются как составные блоки других, более сложных численных методов. В данной лабораторной работе мы рассмотрим одну из старейших задач вариационного исчисления: задачу о брахистохроне, т.е. задачу о кривой наискорейшего спуска. Она состоит в нахождении такой кривой, по которой материальная точка из точки $(x, y) = (0, 0)$ достигнет точки $(x, y) = (a, y_a)$ под действием силы тяжести за наименьшее время (здесь и далее ось y направлена вниз). Решением этой задачи является такая кривая $y(x)$, которая минимизирует следующий функционал, являющийся полным временем движения материальной точки:

$$J[y] = \int_0^a \sqrt{\frac{1+(y'(x))^2}{2gy(x)}} dx, \quad (1)$$

Где g обозначает ускорение свободного падения, и $y'(x) = dy/dx$.

Представленная задача имеет аналитическое решение, которым является параметрически заданная циклоида:

$$\begin{bmatrix} x(t) \\ y(t) \end{bmatrix} = C \begin{bmatrix} t - \frac{1}{2} \sin(2t) \\ \frac{1}{2} - \frac{1}{2} \cos(2t) \end{bmatrix}, \quad (2)$$

где $t \in [0; T]$ и C, T являются константами, значения которых находятся из граничного условия.

В базовой части задания требуется воспользоваться численным интегрированием для нахождения полного времени движения материальной точки по кривой наискорейшего спуска. В продвинутой части требуется разработать метод для нахождения аппроксимации этой кривой. Здесь и далее принимается $a = 2$ и $y_a = 1$. Константы циклоиды для этого граничного условия равны $C = 1.03439984$, $T = 1.75418438$.

Требуется (базовая часть):

1. Разработать функции `composite_simpson(a, b, n, f)` и `composite_trapezoid(a,`

`b, n, f)` численного интегрирования некоторой функции f на интервале $[a; b]$ по

n узлам с помощью составной формулы Симпсона и составной формулы трапеций соответственно.

2. Рассчитать интеграл (1) для функции $y(x)$, соответствующей кривой наискорейшего спуска, с помощью составной формулы Симпсона и составной формулы трапеций для множества значений $n \in [3; 9999]$.

3. На одной координатной плоскости постройте log-log графики зависимостей абсолютной погрешности численного интегрирования от шага интегрирования для обеих формул.

4. Объясните, каким образом по полученному графику можно определить порядок точности формулы.

5. Для обеих формул сравните известные аналитические порядки точности с порядками точности, получаемыми с помощью соответствующих графиков. Обоснуйте алгоритм сравнения.

6. Существует ли оптимальный шаг интегрирования для первой или второй формулы, минимизирующий достижимую погрешность? Обоснуйте свой ответ.

Требуется (продвинутая часть):

1. Используя кусочно-линейную интерполяцию с равноудалёнными узлами, преобразовать задачу о минимизации функционала (1) к полудискретной форме, где аргументами минимизации будут параметры кусочно-линейной интерполяции.

2. Далее, используя составную формулу Симпсона, преобразовать задачу к полностью дискретной форме.

3. Решить полученную задачу минимизации, используя различные конфигурации дискретизации: с шагом интерполяции и шагом интегрирования от 10^{-3} до 1.

4. Используя log-log графики и линии уровня, оценить зависимость погрешности решения от шага интерполяции и шага интегрирования.

5. Опциональное задание №1. Использовать метод градиентного спуска и автоматическое дифференцирование для численного решения задачи оптимизации.

6. Опциональное задание №2. Использовать интерполяцию кубическими сплайнами (возможно, с модификациями) вместо кусочно-линейной интерполяции.

Цель выполнения лабораторной работы

Целью лабораторной работы является изучение алгоритмов численного интегрирования, в частности, составных методов Симпсона и трапеций, определение порядков точности разных методов и оптимального шага для них, а также изучение реализации кусочно-линейной интерполяции и осуществление минимизации, аргументами которой являются коэффициенты прямых кусочно-линейного сплайна.

1. Базовая часть

1.1. Разработка функций численного интегрирования с помощью составной формулы Симпсона и составной формулы трапеций.

Функция для численного интегрирования, использующая составную формулу Симпсона, представлена в листинге 1.

Составные формулы используют кусочный подход для численного интегрирования функций, так как увеличение порядка формул Ньютона-Котеса будет приводить к «паразитным погрешностям» (в основе лежит полином Лагранжа). Эти методы разбивают область интегрирования на некоторое количество подотрезков и аппроксимируют интеграл на каждом.

метод численного интегрирования с помощью составной формулы трапеций, заключается в замене на каждом элементарном отрезке подынтегральной функции на многочлен первой степени, то есть линейную функцию. Площадь под графиком функции аппроксимируется прямоугольными трапециями. Метод Симпсона аппроксимирует интеграл на каждом подотрезке с помощью параболы, проходящей через три точки функции на этом подотрезке, то есть подынтегральная функция заменяется на многочлен 2 степени.

Составная формула Симпсона имеет вид:

$$\int_a^b f(x)dx = \frac{h}{3} \left(f(x_1) + 2 \sum_{i=1}^{\frac{n}{2}-1} f(x_{2i+1}) + 4 \sum_{i=1}^{\frac{n}{2}} f(x_{2i}) + f(x_{n+1}) \right) - \frac{(b-a)h^4}{180} f^{(4)}(\xi), \quad (3)$$

где $x_i = a + (i-1)h$, $h = \frac{b-a}{n}$, $i = 1, \dots, n+1$, $\xi \in (a; b)$, $f(x) \in C^4[a; b]$,

n – число подотрезков на интервале $[a; b]$, должно быть четное, так как количество узлов нечетное,

h – шаг интегрирования,

x_i – узлы,

$\frac{(b-a)h^4}{180}f^{(4)}(\xi)$ – остаточный член, порядок точности составной формулы Симпсона $O(h^4)$.

Листинг 1: функция численного интегрирования с помощью составной формулы Симпсона

```
def composite_simpson(a, b, n, f):  
    if n % 2 != 0:  
        raise ValueError  
  
    x = np.linspace(a, b, n+1)  
    h = (b-a)/n  
  
    integral_value = h/3.*(f(x[0]) + f(x[-1]))  
    integral_value += h/3.*(2*np.sum([f(x_i) for x_i in x[2:-1:2]]))  
    integral_value += h/3.*(4*np.sum([f(x_i) for x_i in x[1::2]]))  
    return integral_value
```

Составная формула трапеций имеет вид:

$$\int_a^b f(x)dx = \frac{h}{2} \left(f(x_1) + 2 \sum_{i=2}^n f(x_i) + f(x_{n+1}) \right) - \frac{(b-a)h^2}{12} f''(\xi), \quad (4)$$

где $x_i = a + (i-1)h$, $h = \frac{b-a}{n}$, $i = 1, \dots, n+1$, $\xi \in (a; b)$, $f(x) \in C^2[a; b]$

n – число подотрезков на интервале $[a; b]$, $n \in \mathbb{N}$,

h – шаг интегрирования,

x_i – узлы,

$\frac{(b-a)h^2}{12}f''(\xi)$ – остаточный член, порядок точности составной формулы трапеций $O(h^2)$.

Функция численного интегрирования, использующая формулу трапеций, представлена в листинге 2.

Обе функции (с использованием формулы Симпсона и с использованием формулы трапеций) имеют одинаковую сигнатуру: они принимают на вход границы интегрирования, заданное количество подотрезков, и подынтегральное выражение.

Листинг 2: функция численного интегрирования с помощью составной формулы трапеций

```
def composite_trapezoid(a, b, n, f):  
    h = (b-a) / n  
    integral_value = h / 2. * (f(a) + f(b))  
  
    for i in range(2, n+1):  
        integral_value += h / 2. * 2 * f(a + (i-1) * h)  
    return integral_value
```

1.2. Расчёт интеграла (1) для функции $y(x)$, соответствующей кривой наискорейшего спуска, с помощью составной формулы Симпсона и составной формулы трапеций.

Так как интеграл (1) требует зависимости $y(x)$, а сама функция, являющаяся решением задачи минимизации полного времени движения материальной точки, задана параметрически, при этом не представляется возможным из параметрического задания получить зависимость $y(x)$ аналитическим образом, то требуется произвести некоторую замену в интеграле и перейти от интегрирования по x к интегрированию по t .

$$dx = d\left(C\left(t - \frac{1}{2}\sin(2t)\right)\right) = C * (1 - 2\cos(2t))dt \quad (5)$$

$$dy = d\left(C\left(\frac{1}{2} - \frac{1}{2}\cos(2t)\right)\right) = C\sin(2t)dt \quad (6)$$

$$y'(t) = \frac{dy}{dx} = \frac{dy}{dt} \frac{dt}{dx} = \frac{\sin(2t)}{1 - \cos(2t)} \quad (7)$$

Подставим полученные выражения (5), (6), (7) в интеграл (1). Получим:

$$\begin{aligned} J[y] &= \int_0^a \sqrt{\frac{1+(y'(x))^2}{2gy(x)}} dx = \int_0^T \sqrt{\frac{1+(y'(t))^2}{2gy(t)}} C * (1 - 2\cos(2t))dt = \\ &= \int_0^T \sqrt{\frac{1+\left(\frac{\sin(2t)}{1-\cos(2t)}\right)^2}{2gC\left(\frac{1}{2}-\frac{1}{2}\cos(2t)\right)}} C * (1 - 2\cos(2t))dt = \\ &= \int_0^T \sqrt{\frac{(1-2\cos(2t)+\cos^2(2t)+\sin^2(2t))C}{g(1-\cos(2t))}} dt = \int_0^T \sqrt{\frac{(2-2\cos(2t))C}{g(1-\cos(2t))}} dt = \int_0^T \sqrt{\frac{2C}{g}} dt = \sqrt{\frac{2C}{g}} t \Big|_0^T \quad (8) \end{aligned}$$

Чтобы найти верхний предел интегрирования по t , найдем t из функции $x(t)$, при этом x следует взять равный $a = 2$ (из условия задачи), что соответствовало верхнему пределу по x .

$$\begin{aligned} x &= t - \frac{1}{2}\sin(2t), \quad x = a = 2 \\ 2 &= \left(t - \frac{1}{2}\sin(2t)\right) \end{aligned} \quad (9)$$

Решив уравнение (9), мы получим, что $t = 1.75418438$. Значит верхний предел интегрирования по t равен константе T из условия задачи.

Рассчитаем значение из формулы (8), полученной аналитическим образом:

$$(10)$$

$$\int_0^T \sqrt{\frac{2C}{g}} dt = \sqrt{\frac{2C}{g}} t \Big|_0^T = \sqrt{\frac{2C}{g}} T \approx 0.805974723403824$$

В листинге 3 записана подынтегральная функция (11), полученная с помощью замены переменной в интеграле (1), в результате чего был произведен переход к интегрированию по t .

$$\int_0^T \sqrt{\frac{1 + (y'(t))^2}{2gy(t)}} C * (1 - 2 \cos(2t)) dt \quad (11)$$

Листинг 3: подынтегральная функция (11)

```
def functional(t):
    C = 1.03439984
    g = 9.80665
    yt = lambda t: C * (0.5 - 0.5 * np.cos(2*t))
    dy_dx = lambda t: np.sin(2*t) / (1 - np.cos(2*t))
    dx_dt = lambda t: C * (1 - np.cos(2*t))

    return np.sqrt((1 + dy_dx(t) ** 2) / yt(t)) * dx_dt(t) / np.sqrt(2 * g)
```

Перейдя к зависимости от t , можно передать функцию из листинга 3 в качестве аргумента для ранее написанных функций численного интегрирования (составных методов Симпсона и трапеций). Сравнивая значения, получаемые с помощью численных методов, с аналитическим решением (10), получим значения абсолютных погрешностей.

Функция (листинг 4) принимает в качестве аргумента метод численного интегрирования, с помощью которого следует делать вычисления. Вычисляя значение функционала (1) для n от 4 до 10000 с шагом 4, функция сопоставляет результат со своим n , тем самым формируя массив значений для каждого количества подынтервалов интегрирования. В качестве нижнего предела интегрирования вместо 0 взято значение 10^{-7} , так как при t , равном нулю, $y(t)$, находящаяся в знаменателе (см. формулу 11), также равна нулю, то есть $t = 0$ – критическая точка.

Листинг 4: функция вычисления значений функционала (1) для разных шагов интегрирования

```
def count_functional(method):
    values = [[], []]
    C = 1.03439984
    T = 1.75418438

    for n in range(4, 10000, 4):
        values[0].append(n)
        values[1].append(method(1e-7, 1.75418438, n, functional))
    return values
```

1.3. Построение графика зависимости абсолютных погрешностей численного интегрирования в зависимости от шага интегрирования.

Получив массив значений функционала (1) с помощью функции из листинга 4, можно найти погрешность относительного аналитического решения (10). Данную задачу выполняет функция (листинг 5). Она получает на вход результат работы функции из листинга 4 (ранее описанный массив), находит шаг интегрирования, поделив длину отрезка интегрирования на количество подынтервалов интегрирования n и сопоставляет с значением погрешности, путем вычисления модуля разности результата численного интегрирования и точного значения функционала (точное значение является входным параметром функции), отображает координаты на log-log графике, абсцисса – шаг интегрирования, ордината – абсолютная погрешность. Помимо прочего функции передаются легенда для графика и цвет для точек этого графика. Результат работы отображен на рис. 1.

Листинг 5: функция нахождения абсолютных погрешностей

```
def plot_inaccuracy(values, precise, color, label):
    integration_steps = []
    integral_values = []
    C = 1.03439984
    T = 1.75418438

    for i in range(len(values[0])):
        integration_steps.append(T/values[0][i])
        integral_values.append(abs(precise - values[1][i]))

    plt.loglog(integration_steps, integral_values, 'o', color=color,
label=label)
```

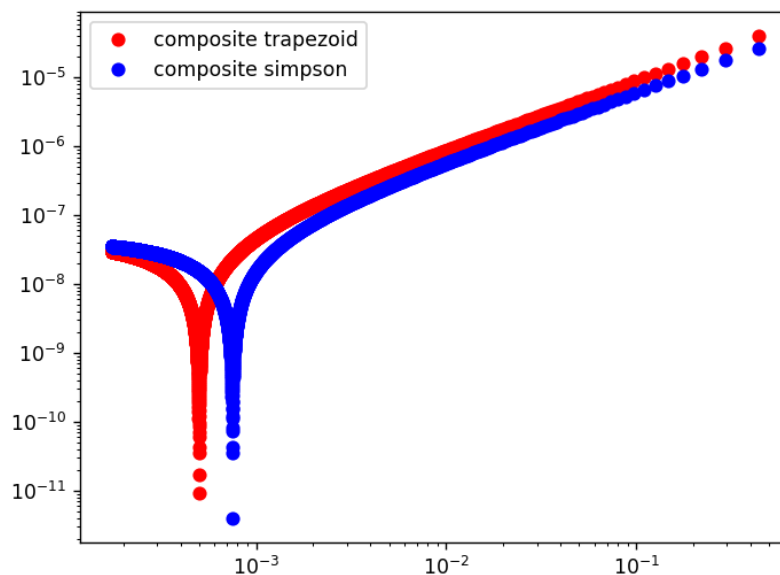


Рис. 1—график зависимости абсолютных погрешностей численного интегрирования функционала от шага интегрирования

1.4. Определение порядка точности формул по графику.

Для проверки правильности работы функций, реализующих методы численного интегрирования, полученные ранее, вычислим с помощью них значения определенного интеграла от экспоненты (например, на интервале от 1 до 5), так как она является гладкой функцией порядка ∞ или бесконечно-дифференцируемой функцией, то есть она, имеет производные всех порядков. Возьмем количество подынтервалов интегрирования n от 4 до 10000. С помощью функции из листинга 5 отобразим абсолютные погрешности. Результат на рис. 2. Для оценки порядка точности, построим на графике функции $E_1(h) = h^2$ и $E_2(h) = h^4$, которые соответствуют 2 и 4 порядку точности. Как можно заметить, при уменьшении h результат метода трапеций убывает пропорционально h^2 , а результат метода Симпсона – пропорционально h^4 (соответствующие «прямые» на log-log графике параллельны). Более строго говоря, при $h \rightarrow 0$ функции абсолютной погрешности от шага интегрирования для метода Симпсона одного порядка малости с $E_2(h) = h^4$, а метод трапеций – с $E_1(h) = h^2$, то есть эти функции отличаются на константный множитель и значения функций и ее производных стремятся к нулю с одинаковой скоростью.

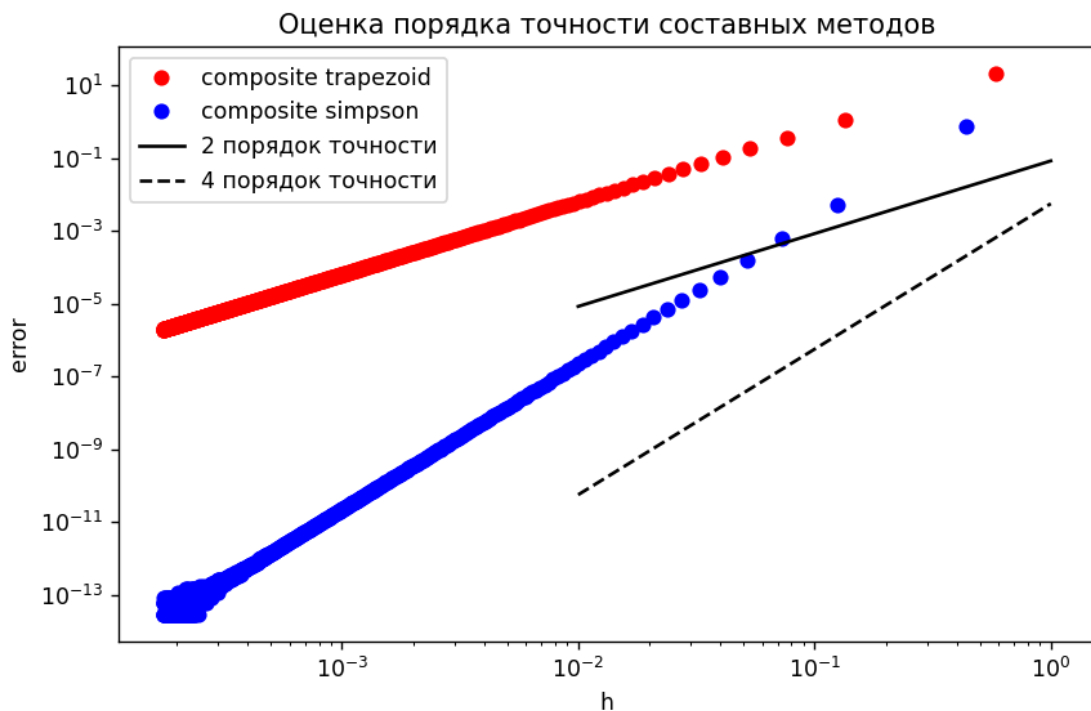


Рис. 2 — оценка порядка точности формул численного интегрирования экспоненты

1.5. Сравнение известных аналитических порядков точности с порядками точности, получаемыми с помощью соответствующих графиков.

Остаточный член составной формулы Симпсона имеет следующий вид:

$$-\frac{(b-a)h^4}{180}f^{(4)}(\xi),$$

где $\xi \in (a; b)$, $f(x) \in C^4[a; b]$

Следовательно аналитический порядок точности $O(h^4)$.

Остаточный член составной формулы трапеций имеет следующий вид:

$$-\frac{(b-a)h^2}{12}f''(\xi),$$

где $\xi \in (a; b)$, $f(x) \in C^2[a; b]$.

Следовательно аналитический порядок точности $O(h^2)$.

Данные порядки точности совпадают с порядками точности, полученными графически на рис. 2, на основе угла наклона прямых на log-log графике.

Определим порядок точности для графиков зависимости абсолютных погрешностей численного интегрирования функционала (1) в зависимости от шага интегрирования.

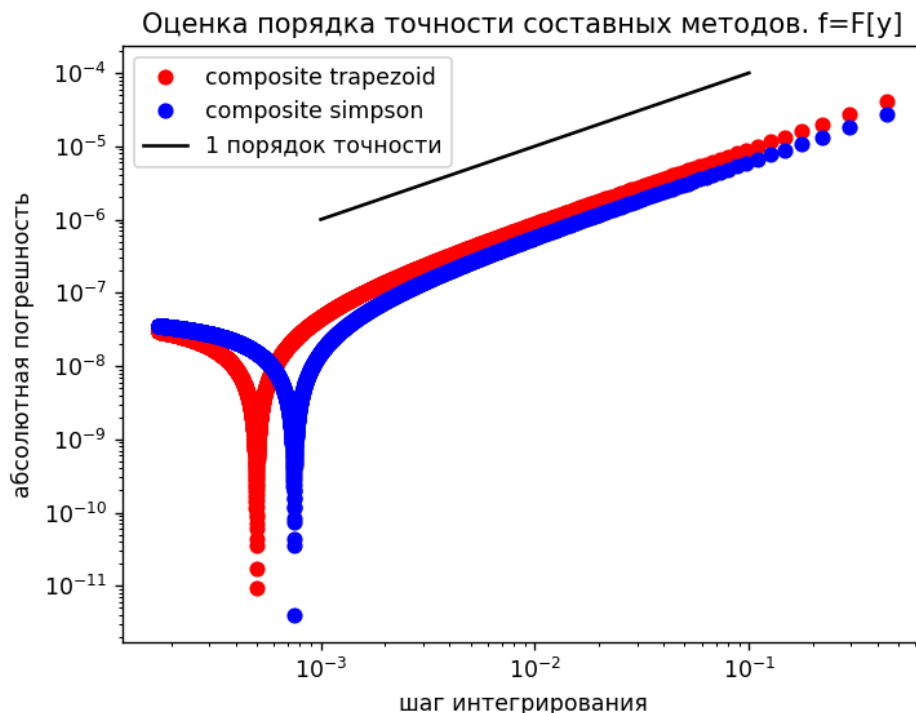


Рис. 3 — график зависимости абсолютных погрешностей численного интегрирования в зависимости от шага интегрирования и прямая $E_3(h)=h$, отображающая угол наклона для 1 порядка точности

В данном случае обе формулы имеют 1 порядок точности, что видно, если построить $E_3(h) = h$ на log-log графике. Это связано с тем, что нарушается

необходимое условие непрерывной дифференцируемости (11) ($t = 0$ – критическая точка).

1.6. *Существование оптимального шага интегрирования для первой и второй формулы, минимизирующего достижимую погрешность.*

Операция интегрирования обладает вычислительной устойчивостью. Интуитивное объяснение этого эффекта заключается в том, что интегрирование предполагает суммирование близких значений, в то время как, например, дифференцирование, являющееся вычислительно неустойчивым, вычисляет их разность. Поэтому полная погрешность интегрирования падает до тех пор, пока она не достигнет значения, сравнимого с машинным эpsilon, после чего уменьшение погрешности становится невозможным, и она стабилизируется на уровне машинного эpsilon. Следовательно, оптимальным можно считать максимальный шаг интегрирования, при котором погрешность сопоставима с машинным эpsilon, поскольку при дальнейшем уменьшении шага будет увеличиваться время работы программы, но абсолютная погрешность уменьшаться уже не будет. Таким образом, для интегрирования экспоненты методом Симпсона, оптимальным является шаг примерно $h = 4 \cdot 10^{-4}$, далее погрешность перестает убывать. Для интегрирования экспоненты методом трапеций, даже шаг $h = 10^{-4}$ дает ошибку много больше, чем машинный эpsilon. Для функционала (1) оптимальным шагом, судя по графику, можно считать $h = 8 \cdot 10^{-4}$ для формулы Симпсона, для формулы трапеций – примерно $h = 5 \cdot 10^{-4}$.

2. Продвинутая часть

2.1. *Преобразование задачи о минимизации функционала к полудискретной форме с помощью кусочно-линейной интерполяции.*

Аппроксимируем функционал (1) с помощью кусочно-линейной интерполяции. Линейная функция на i -ом подынтервале будет выглядеть следующим образом:

$$l_i(x) = k_i(x - x_i) + b_i = \frac{y_{i+1} - y_i}{x_{i+1} - x_i}(x - x_i) + y_i, \quad (12)$$

где k_i – угловой коэффициент прямой, b_i – свободный член, y_i и x_i – интерполяционные узлы, $x \in [x_i, x_{i+1}]$, $i = 0, 1 \dots N - 1$, N – количество интервалов кусочной интерполяции.

Найдем первообразную функции (12):

$$\int l_i(x) dx = \int k_i(x - x_i) + b_i = \frac{k_i}{2}(x - x_i)^2 + b_i x \quad (13)$$

Заменим функцию на ее интерполяцию:

$$\int_0^a \sqrt{\frac{1 + (y'(x))^2}{2gy(x)}} dx = \sum_{i=0}^{N-1} \int_{x_i}^{x_{i+1}} \sqrt{\frac{1 + (l'_i(x))^2}{2gl_i(x)}} dx \quad (14)$$

Получена полудискретная форма (13) функционала (1).

В листинге 6 представлена программная реализация функции вычисления коэффициентов k_i и b_i кусочно-линейного интерполянта. Задаётся количество подынтервалов интерполяции, затем коэффициенты высчитываются по координатам узлов интерполяции, как это показано в (12), после чего заносятся в массив и возвращаются.

Листинг 6: вычисление коэффициентов кусочно-линейного сплайна

```
def get_coefs_linear_spline(x_nodes, y_nodes, N):
    coefs = [[], []]
    for j in range(N):
        slope = (y_nodes[j + 1] - y_nodes[j]) / (x_nodes[j + 1] - x_nodes[j])
        intercept = y_nodes[j]
        coefs[0].append(slope)
        coefs[1].append(intercept)
    return coefs
```

2.2. Преобразование функционала к полностью дискретной форме с помощью составной формулы Симпсона.

Преобразуем интеграл (14) с использованием составной формулы Симпсона, приведя функционал, таким образом, к полностью дискретной форме:

$$\sum_{i=0}^{N-1} \int_{x_i}^{x_{i+1}} \sqrt{\frac{1 + (l'_i(x))^2}{2gl_i(x)}} dx = \sum_{i=0}^{N-1} \frac{h}{3} [l_i(x_1) + 2 \sum_{j=1}^{\frac{n}{2}-1} l_i(x_{2j+1}) + 4 \sum_{j=1}^{\frac{n}{2}} l_i(x_{2j}) + l_i(x_{n+1})], \quad (15)$$

где n – количество подынтервалов интегрирования.

После проделанных преобразований можно вычислить значение функционала. Функция, совершающая это, указана в листинге 7. Она принимает в качестве аргументов коэффициенты кусочно-линейного сплайна, шаг интегрирования и шаг интерполирования, абсциссы узлов интерполяции. На каждом интервале интерполяции происходит численное интегрирование с помощью составной формулы Симпсона с использованием заданного шага интегрирования, суммируются результаты вычислений интеграла на всех интервалах интерполяции. Но также можно проинтегрировать аналитически, первообразная линейной функции записана в (13), однако, результат будет такой же, потому что составная формула Симпсона имеет вторую степень

точности, то есть 2 степень полинома и менее при интегрировании с помощью данной формулы будет высчитана точно.

Листинг 7: вычисление значения функционала с помощью кусочно-линейной интерполяции и составной формулы Симпсона

```
def functional_value_with_linear_spline(x_nodes, coefs_matrix, N, n_integr):
    integral_value_simpson = 0
    integral_value_F = 0
    def create_linear_f(slope, intercept, x_i):
        linear_f = lambda x: slope * (x-x_i) + intercept
        linear_F = lambda x: slope / 2 * (x-x_i)**2 + intercept * x
        return linear_f, linear_F
    for i in range(N-1):
        f, F = create_linear_f(coefs_matrix[0][i], coefs_matrix[1][i],
x_nodes[i])
        integral_value_simpson += composite_simpson(x_nodes[i], x_nodes[i+1],
n_integr, f)
        integral_value_F += F(x_nodes[i+1]) - F(x_nodes[i])
    return integral_value_simpson, integral_value_F
```

Проверим корректно ли найдены коэффициенты кусочно-линейной интерполяции, для этого напишем функцию построения сплайна (листинг 8). Данная функция получает на вход в качестве аргументов коэффициенты сплайна, узлы интерполяции. Данные значения подставляются в (12), и вычисляются координаты точек прямых кусочной интерполяции. Инвертируется направление оси, так как по условию она направлена вниз. Результат для 10 интервалов интерполяции представлен на рис. 4, для 50 – на рис. 5.

Листинг 8: построение кусочно-линейного сплайна

```
def plot_linear_spline(x_nodes, coefs_matrix, N):
    for i in range(N):
        x = np.linspace(x_nodes[i], x_nodes[i + 1], 2)
        y = coefs_matrix[0][i] * (x - x_nodes[i]) + coefs_matrix[1][i]
        plt.plot(x, y)
    plt.title('кусочно-линейная интерполяция кривой наискорейшего спуска')
    plt.xlabel('x')
    plt.ylabel('y')
    plt.gca().invert_yaxis()
    plt.show()
```

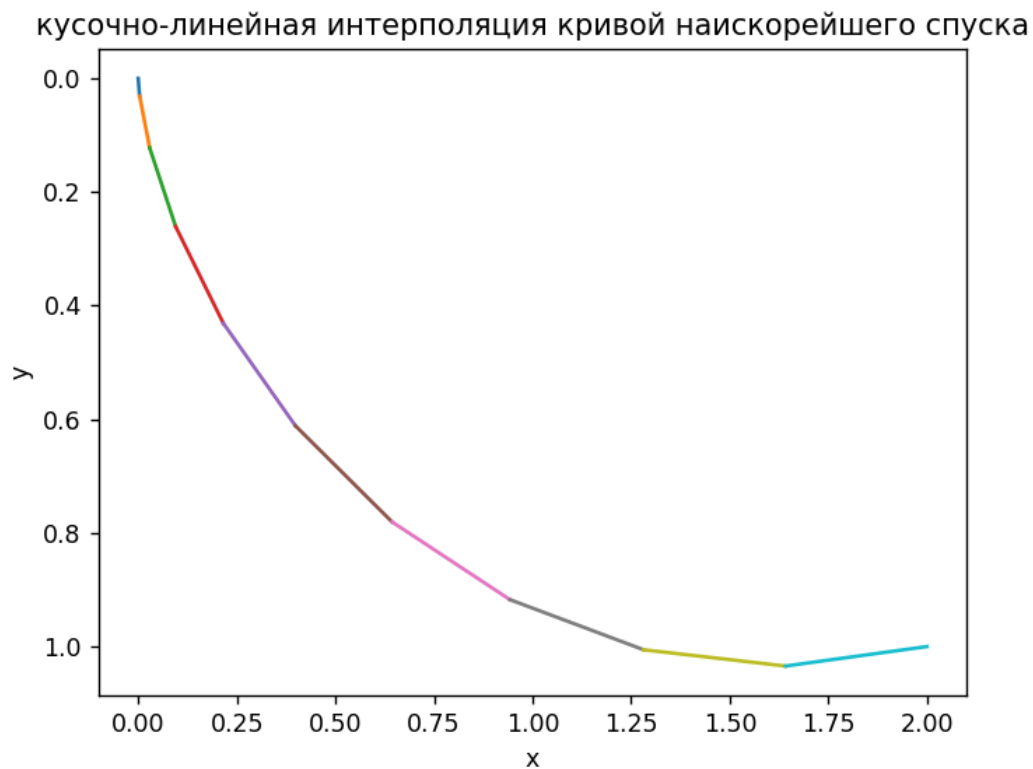


Рис. 4 — график интерполяции кривой наискорейшего спуска кусочно-линейным сплайном для $N = 10$

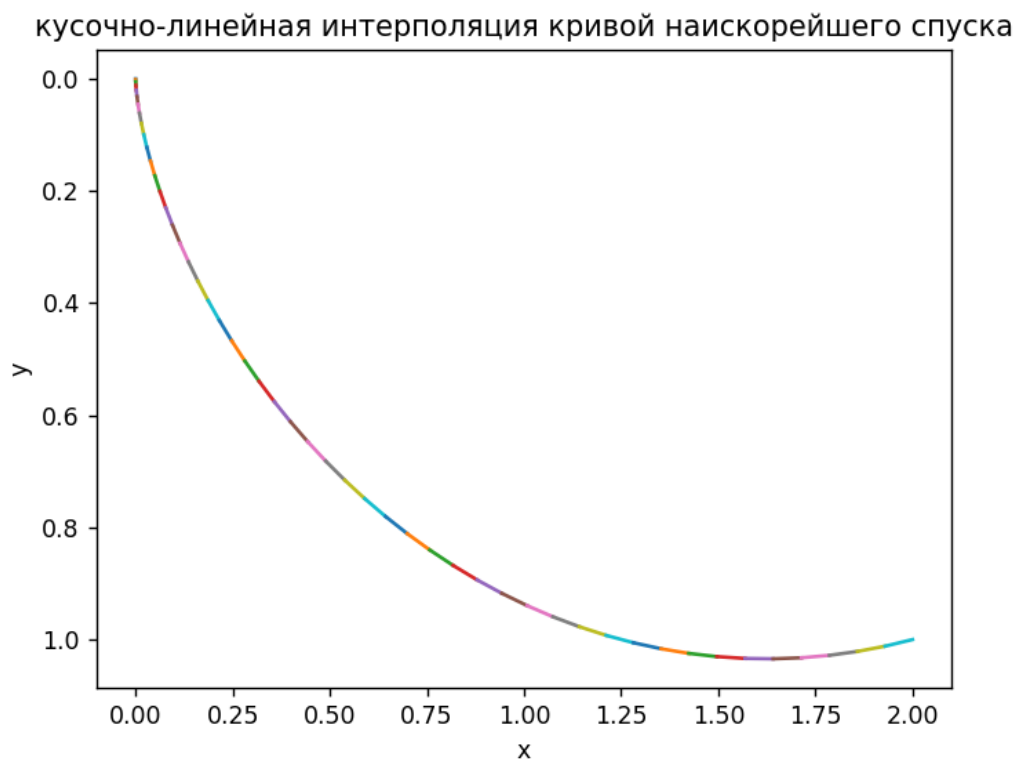


Рис. 5 — график интерполяции кривой наискорейшего спуска кусочно-линейным сплайном для $N = 50$

Заключение

В результате выполнения лабораторной работы были реализованы составные методы для численного интегрирования: метод Симпсона и метод

трапеций. С помощью них было получено численное значение функционала полного времени движения материальной точки по кривой наискорейшего спуска. После этого были вычислены абсолютные погрешности для разных шагов интегрирования. Оценив порядок точности каждой из формул, для нескольких функций, можно сделать вывод о важности условия гладкости интегрируемой функции, так как при невыполнении этого условия, порядок точности снижается. Если данное условие выполняется, то фактический порядок точности совпадает с аналитическим. В продвинутой части лабораторной работы функционал был приведён сначала к полудискретной форме, путём замены подынтегрального выражения кусочно-линейным сплайном, а затем к полностью дискретной форме путём аппроксимации интеграла составной формулой Симпсона. Была сформирована программная реализация нахождения значения функционала, использующая его дискретную форму.

Список использованных источников

1. Першин А.Ю. Лекции по курсу «Вычислительная математика». Москва, 2018- 2021. С. 140. URL: <https://archrk6.bmstu.ru/index.php/f/810046>.
2. Соколов, А.П. Инструкция по выполнению лабораторных работ (общая). Москва: Соколов, А.П., 2018-2021. С. 9. URL: <https://archrk6.bmstu.ru>. (облачный сервис кафедры РК6).
3. Соколов, А.П. Инструкция по выполнению заданий к семинарским занятиям (общая). Москва: Соколов, А.П., 2018-2022. С. 7. URL: <https://archrk6.bmstu.ru>. (облачный сервис кафедры РК6).
4. Першин А.Ю. Сборник задач семинарских занятий по курсу «Вычислительная математика»: Учебное пособие. / Под редакцией Соколова А.П. [Электронный ресурс]. Москва, 2018-2021. С. 20. URL: <https://archrk6.bmstu.ru>. (облачный сервис кафедры РК6).
5. Першин А.Ю., Соколов А.П. Сборник постановок задач на лабораторные работы по курсу «Вычислительная математика»: Учебное пособие. [Электронный ресурс]. Москва, 2021. С. 54. URL: <https://archrk6.bmstu.ru>. (облачный сервис кафедры РК6)