



Министерство науки и высшего образования Российской Федерации
федеральное государственное бюджетное образовательное учреждение
высшего профессионального 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:	Очиров Бадма Юрьевич
Группа:	РК6-55Б
Тип задания:	лабораторная работа
Тема:	Использование аппроксимаций для численной оптимизации

Студент

подпись, дата

Очиров Б. Ю.

Фамилия, И.О.

Преподаватель

подпись, дата

Соколов А. П.

Фамилия, И.О.

Москва, 2021

Содержание

Использование аппроксимаций для численной оптимизации	3
1 Задание	3
2 Цель выполнения лабораторной работы	4
3 Базовая часть	5
4 Продвинутая часть	22
5 Заключение	26

Использование аппроксимаций для численной оптимизации

1 Задание

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:

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

где g обозначает ускорение свободного падения, и $y'(x) = dy/dx$. Эта 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 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 $C = 1.03439984$, $T = 1.75418438$.

Базовая часть:

1. Написать функцию `composite_simpson(a, b, n, f)` численного интегрирования функции f на интервале $[a; b]$ по n узлам с помощью составной формулы Симпсона.
2. Написать функцию `composite_trapezoid(a, b, n, f)` численного интегрирования функции f на интервале $[a; b]$ по n узлам с помощью составной формулы трапеций.
3. Рассчитать интеграл (1) для функции $y(x)$, соответствующей кривой наискорейшего спуска, с помощью составной формулы Симпсона и составной формулы

трапеций для множества значений $n \in [3; 9999]$. Постройте $\log\text{-}\log$ график зависимости 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 порядок, полученный с помощью графика, с аналитическим порядком точности.
6. Существует ли оптимальный шаг интегрирования для данной формулы, минимизирующий достижимую погрешность? Обоснуйте свой ответ.

Продвинутая часть:

1. Используя кусочно-линейную интерполяцию с равноудаленными 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 $\log\text{-}\log$ 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.

2 Цель выполнения лабораторной работы

Цель выполнения лабораторной работы – изучить методы численного интегрирования и использовать их для нахождения полного времени движения материальной точки по кривой наискорейшего спуска, разработать метод для нахождения аппроксимации этой кривой преобразованием задачи о минимизации функционала (1) к полудискретной и дискретной формам.

3 Базовая часть

1. Разработка функции численного интегрирования с помощью составной формулы Симпсона.
2. Разработка функции численного интегрирования с помощью составной формулы трапеций.
3. Расчёт интеграла (1) для функции $y(x)$ с помощью составной формулы Симпсона/трапеций. Построение log-log графика зависимости абсолютной погрешности численного интегрирования от шага интегрирования для обеих формул.
4. Определение порядка точности формулы по полученному графику.
5. Сравнение порядка точности, полученный с помощью графика, с аналитическим порядком точности для обеих формул.
6. Анализ существования оптимального шага интегрирования для данной формулы, минимизирующий достижимую погрешность.

1. Разработка функции `composite_simpson(a, b, n, f)`.

Хорошо, что есть ссылка на лекции

Замечание: в лекциях формулы указаны для $n+1$ узлов, в данной работе формулы переписаны для n узлов в соответствии с условием лабораторной работы.

Пусть $x_i = a + (i-1) \cdot h$, $h = \frac{b-a}{n-1}$ и $i = 1, \dots, n$, где n - нечетное количество узлов ($n \neq 1$). Тогда существует такое $\xi \in (a; b)$ для $f(x) \in C^4[a; b]$, что составная формула Симпсона имеет вид:

Хотя ссылка на лекции имеется выше, стоило все же пояснить, что это за обозначения, но это, скорее, общее замечание

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

В реализации для случая, когда количество узлов чётно, инкрементируем его на единицу, чтобы всё вычислялось для любого $n \in N$ ($n \neq 1$).

Для реализации и вычислений используется только левая часть формулы (3):

$$\int_a^b f(x)dx \approx \frac{h}{3} \cdot \left[f(x_1) + 2 \cdot \sum_{i=1}^{(n-3)/2} f(x_{2i+1}) + 4 \cdot \sum_{i=1}^{(n-1)/2} f(x_{2i}) + f(x_n) \right]; \quad (4)$$

Хорошо, что здесь не знак равенства

Скорее, "реализующая"

Функция `composite_simpson(a, b, n, f)`, реализовывающая формулу (4), представлена в Листинге 1.

Листинг 1. Реализация функции `composite_simpson(a, b, n, f)`

```
1 def composite_simpson(a, b, n, f):
2     n += (n + 1) % 2
3
4     pr = n - 1
5     h = (b - a) / pr
6     x = a + h #x_2
7
8     sum1 = 0
9     sum2 = f(x) #f(x_2)
10    for i in range(1, pr // 2):
11        x += 2 * h
12        sum1 += f(x - h) #f(x_3), f(x_5), ...
13        sum2 += f(x) #f(x_4), f(x_6), ...
14
15    result = (h / 3) * (f(a) + 2 * sum1 + 4 * sum2 + f(b))
16
17    return result
```

Комментарии в листинге в целом позволяют разобраться, что происходит, но стоило все же выше кратко описать реализацию данной функции

2. Разработка функции `composite_trapezoid(a, b, n, f)`.

Пусть $x_i = a + (i - 1) \cdot h$, $h = \frac{b - a}{n - 1}$ и $i = 1, \dots, n$, где $n \in \mathbb{N}$ ($n \neq 1$). Тогда существует такое $\xi \in (a; b)$ для $f(x) \in C^2[a; b]$, что составная формула трапеций имеет вид:

Здесь уже нет ссылки на лекции

$$\int_a^b f(x) dx = \frac{h}{2} \cdot \left[f(x_1) + 2 \cdot \sum_{i=2}^{n-1} f(x_i) + f(x_n) \right] - \frac{(b - a) \cdot h^2}{12} f''(\xi); \quad (5)$$

Для реализации и вычислений используется только левая часть формулы (5):

$$\int_a^b f(x) dx = \frac{h}{2} \cdot \left[f(x_1) + 2 \cdot \sum_{i=2}^{n-1} f(x_i) + f(x_n) \right]; \quad (6)$$

Здесь все же знак примерного равенства

Функция `composite_trapezoid(a, b, n, f)`, реализовывающая формулу (6), представлена в Листинге 2.

Листинг 2. Реализация функции `composite_trapezoid(a, b, n, f)`

```
1 def composite_trapezoid(a, b, n, f):
2     pr = n - 1
3
4     h = (b - a) / pr
5     x = a
6
7     sum0 = 0
```

```

8   for i in range(2, pr + 1):
9       x += h
10      sum0 += f(x)
11
12      result = (h / 2) * (f(a) + 2 * sum0 + f(b))
13
14      return result

```

3. Расчёт интеграла (1) для функции $y(x)$ с помощью составной формулы Симпсона/трапеций. Построение log-log графика зависимости абсолютной погрешности численного интегрирования от шага интегрирования для обеих формул.

Используя выражение (2), можно найти зависимость t от y :

$$\begin{aligned}
 y(t) &= \frac{1}{2} \cdot C \cdot (1 - \cos(2t)); \\
 \frac{2 \cdot y}{C} &= 1 - \cos(2t); \\
 \frac{2 \cdot y}{C} &= 2 \cdot \sin^2(t); \\
 \frac{y}{C} &= \sin^2(t) = 1 - \cos^2(t);
 \end{aligned} \tag{7}$$

Для выражения (7) получим следующее:

$$\begin{aligned}
 \sin^2(t) &= \frac{y}{C}; & \cos^2(t) &= 1 - \frac{y}{C}; \\
 \sin(t) &= \left| \sqrt{\frac{y}{C}} \right|; & \cos(t) &= \left| \sqrt{1 - \frac{y}{C}} \right|; \\
 \sin(t) &= \sqrt{\frac{y}{C}}, & t &= \arccos\left(\left| \sqrt{1 - \frac{y}{C}} \right| \right),
 \end{aligned} \tag{8}$$

где $0 \leq y \leq C$. Знак, с которым раскрывается модуль при синусе, является положительным на рассматриваемом отрезке $t \in [0; T]$. Знак, с которым раскрывается модуль при косинусе, зависит от $t : t \in [0; \frac{\pi}{2}] \rightarrow$ со знаком '+', $t \in [\frac{\pi}{2}; T] \rightarrow$ со знаком '-'.

Подставим выражения (8) в (2) для $x(t)$:

$$x(t) = C \cdot \left(t - \frac{1}{2} \cdot \sin(2t) \right) = C \cdot (t - \sin(t) \cdot \cos(t));$$

Таким образом, получена зависимость x от y :

$$x(y) = C \cdot \left(\arccos\left(\left| \sqrt{1 - \frac{y}{C}} \right| \right) - \sqrt{\frac{y}{C}} \cdot \left| \sqrt{1 - \frac{y}{C}} \right| \right);$$

Найдём производные x и y по t :

$$x'(t) = \frac{dx}{dt} = C \cdot (1 - \cos(2t)) = 2 \cdot C \cdot \sin^2(t); \quad (9)$$

$$y'(t) = \frac{dy}{dt} = C \cdot \sin(2t) = 2 \cdot C \cdot \sin(t) \cdot \cos(t); \quad (10)$$

С помощью выражений (9) и (10) найдена производная y по x :

$$y'(x) = \frac{dy}{dx} = \frac{2 \cdot C \cdot \sin(t) \cdot \cos(t) \cdot dt}{2 \cdot C \cdot \sin^2(t) \cdot dt} = \frac{\cos(t)}{\sin(t)} = \frac{\left| \sqrt{1 - \frac{y}{C}} \right|}{\sqrt{\frac{y}{C}}} = \frac{\left| \sqrt{C - y} \right|}{\sqrt{y}},$$

где $\sin(t) \neq 0 \Rightarrow t \neq 0$, знак модуля положительный для $y \in (0; C]$ и отрицательный для $y \in [C; 1]$.

Преобразуем интеграл (1):

$$\begin{aligned} \mathcal{F}[y] &= \int_0^a \sqrt{\frac{1 + (y'(x))^2}{2 \cdot g \cdot y(x)}} dx = \int_0^a \sqrt{\frac{1 + \left(\frac{\left| \sqrt{C - y} \right|}{\sqrt{y}} \right)^2}{2 \cdot g \cdot y}} dx \\ &= \int_0^a \sqrt{\frac{1 + \frac{C - y}{y}}{2 \cdot g \cdot y}} dx = \sqrt{\frac{C}{2 \cdot g}} \cdot \int_0^a \sqrt{\frac{1}{y^2(x)}} dx \\ &= \sqrt{\frac{C}{2 \cdot g}} \cdot \int_0^a \frac{dx}{|y(x)|} = \sqrt{\frac{C}{2 \cdot g}} \cdot \int_0^a \frac{dx}{y(x)}; \end{aligned} \quad (11)$$

Модуль раскрывается с положительным знаком, так как $0 \leq y \leq C$.

Интеграл (11) можно перевести в интегрирование по dt :

$$\begin{aligned} \mathcal{F}[y] &= \sqrt{\frac{C}{2 \cdot g}} \cdot \int_0^T \frac{2 \cdot C \cdot \sin^2(t) \cdot dt}{y(t)} = \sqrt{\frac{C}{2 \cdot g}} \cdot \int_0^T \frac{2 \cdot C \cdot \sin^2(t) \cdot dt}{C \cdot \left(\frac{1}{2} - \frac{1}{2} \cos(2t) \right)} = \\ &= \sqrt{\frac{C}{2 \cdot g}} \cdot \int_0^T \frac{4 \cdot \sin^2(t) \cdot dt}{1 - \cos(2t)} = \sqrt{\frac{C}{2 \cdot g}} \cdot \int_0^T \frac{4 \cdot \sin^2(t) \cdot dt}{2 \cdot \sin^2(t)} = \\ &= \sqrt{\frac{2 \cdot C}{g}} \cdot \int_0^T dt = \sqrt{\frac{2 \cdot C}{g}} \cdot t \Big|_0^T; \end{aligned}$$

Так как известно, что для $t = 0$ производная $y'(x)$ неопределима, то и интегрирование по dt для $t = 0$ некорректно. Соответственно, заменим 0 на некоторую точку t_a ,

близкую к 0, и используем следующую формулу для вычисления точного значения интеграла:

$$\mathcal{F}[y] = \sqrt{\frac{2 \cdot C}{g}} \cdot t \Big|_0^T = \sqrt{\frac{2 \cdot C}{g}} \cdot t \Big|_{t_a}^T = \sqrt{\frac{2 \cdot C}{g}} \cdot (T - t_a); \quad (12)$$

Отлично, что
подробно
описано, как
найденное данное
значение

Также интегрируем (11) по dy :

$$\begin{aligned} \mathcal{F}[y] &= \sqrt{\frac{C}{2 \cdot g}} \cdot \int_0^C \frac{\frac{\sqrt{y}}{|\sqrt{C-y}|} \cdot dy}{y} = \sqrt{\frac{C}{2 \cdot g}} \cdot \left(\int_0^C \frac{\frac{\sqrt{y}}{\sqrt{C-y}} \cdot dy}{y} + \int_C^1 \frac{-\frac{\sqrt{y}}{\sqrt{C-y}} \cdot dy}{y} \right) = \\ &= \sqrt{\frac{C}{2 \cdot g}} \cdot \left(\int_0^C \frac{dy}{\sqrt{C-y} \cdot \sqrt{y}} + \int_1^C \frac{dy}{\sqrt{C-y} \cdot \sqrt{y}} \right) \end{aligned} \quad (13)$$

Да, все верно

Выражение (12) примем за точное значение интеграла для оценки погрешности, а t_a будем менять в зависимости от условий интегрирования. А интегралы (11), (13) будем использовать для численного интегрирования.

Интегрируем по dx (основной подход):

Интегрируем по dx для равномерно распределённых узлов x . Так как $y(x)$ не задан и не выражается явно, то для получения значений y по заданным x используем два метода: **1)** массив из $10^4 + 1$ ($10^5 + 1, 10^6 + 1$) сгенерированного узла, из которого мы подставляем значения в подынтегральное выражение, **2)** метод интерполяции кубическими сплайнами для 1001 узла.

Заметим, что в выражении (11) функция под интегралом зависит от $y(x)$ в знаменателе. Соответственно, при тех x , когда $y(x)$ принимает близкое к 0 значение, подынтегральная функция растёт до бесконечности.

Очень хорошее пояснение

Так как метод численного интегрирования непосредственно состоит в том, чтобы аппроксимировать интеграл квадратурой, то есть в нашем случае с составными формулами мы имеем отрезок $[0; 2]$, который в свою очередь делится на $n - 1$ отрезок, где для каждого такого отрезка используется формула Симпсона/трапеций. В таком случае около 0 происходят следующие вычисления: $\frac{h}{m} \cdot (f(a) + \dots)$, где h - шаг интегрирования, который уменьшается с увеличением числа узлов, m - некоторое число, зависящее от метода интегрирования, а $f(a)$ и другие значения около 0 стремятся к бесконечности. Это приводит к тому, что в зависимости от приближения к 0, мы получаем бесконечно большое значение в точке, в то время как в самой точке $a = 0$ значение $f(a)$ получить невозможно.

Соответственно, приближение к границе слева будем находить логически и экспериментально, добиваясь наименьших погрешностей. Эти выводы справедливы для всех дальнейших вычислений - не только для интегрирования по dx , но и по dy .

Суть первого метода состоит в том, чтобы подставлять в подынтегральную функцию значение y из заранее сгенерированного списка значений y_nodes .

Генерирование происходит в функции *massive_creation(a, b)* для равномерно распределённых узлов от $\frac{1}{L}$ до 2 с шагом $h = \frac{2 - \frac{1}{L}}{L}$, где $L = [10^4, 10^5, 10^6]$. Начало отрезка обусловлено тем, что интегрирование приходится выполнять не от нулевого значения.

Реализация функции простая, она представлена в Листинге 3.

Листинг 3. Реализация функции *massive_creation(a, b)*

```

1 def massive_creation(a, b):
2     global dots, y_nodes
3
4     x_nodes = np.linspace(a, b, dots + 1)
5     y_nodes = []
6     for i in x_nodes:
7         y_nodes.append(y_function_approx(i))

```

Вычисление каждого элемента списка *y_nodes* в функции *massive_creation(a, b)* производится с помощью следующего поискового метода значений *y*:

Известны выражения (2) для $x(t), y(t)$, по ним можно определить соответствие x и y через $t \in [0; T]$. Функция *y_function_approx(x)* принимает значение x , варьирует по алгоритму t от 0 до того момента, пока $x(t)$ не совпадёт с заданным x . Когда это произойдёт, используем значение t для нахождения $y(t)$ Вот здесь уже есть описания листингов. Это хорошо

Реализация данной функции представлена в Листинге 4.

Листинг 4. Реализация функции *y_function_approx(x)*

```

1 def y_function_approx(x):
2     C = 1.03439984
3     eps = 1
4     t = 0
5
6     while eps >= 0.000000000001:
7         while C * (t - (1. / 2) * np.sin(2 * t)) <= x:
8             t += eps
9             t -= eps
10            eps /= 10
11
12    y = C * (1. / 2) * (1 - np.cos(2 * t))
13
14    return y

```

То, каким образом эти значения используются далее, будет рассмотрено после ознакомления со вторым методом определения y .

Второй метод более сложный, он заключается в приближении $y(x)$ кубическими сплайнами с разделением на 1000 отрезков. Для этого потребуются функции *qubic_spline_coeff(x_nodes, y_nodes)* и *qubic_spline(x_nodes, x, qs_coeff)* из первой лабораторной работы, они были немного модифицированы для ускорения вычислений. Эти

Правильная ссылка, но хотелось бы видеть тогда ее и в списке литературы

функции и функция `nodes_creation(a, b)`, которая непосредственно генерирует все коэффициенты кубического сплайна, представлены в файле `basicPart2.ipynb`.

Полный отрезок интегрирования во втором методе рассматривается точно так же, как это сделано в первом методе: от $\frac{1}{L}$ до 2, где $L = [10^4, 10^5, 10^6]$. Он же рассматривается, как отрезок, на котором рассматривается интерполяция функции $y(x)$.

После того, как эти два метода были применены, необходимо численно проинтегрировать интеграл (11):

$$\sqrt{\frac{C}{2 \cdot g}} \cdot \int_{1/L}^2 \frac{dx}{y(x)};$$

Чтобы рассчитать интеграл, используем функции, реализующие составные формулы Симпсона и трапеций, - `composite_simpson(a, b, n, f)`, `composite_trapezoid(a, b, n, f)`. Границы отрезка a и b известны, n - инкрементируемая величина от 3 до 9999, а f - подынтегральная функция:

Реализация подынтегральной функции для **первого метода** в Листинге (5):

Листинг 5. Реализация функции `under_integral_function(x)` для первого метода

```

1 def under_integral_function(x):
2     global dots, y_nodes, ds
3
4     xx = np.around(x, decimals=ds)
5     index = int(np.around(xx / 2 * dots))
6     t = y_nodes[index]
7
8     return 1 / t

```

Реализация подынтегральной функции для **второго метода** в Листинге (6):

Листинг 6. Реализация функции `under_integral_function(x)` для второго метода

```

1 def under_integral_function(x):
2     global x_nodes, qs_coeff
3
4     return 1 / qubic_spline(x_nodes, x, qs_coeff)

```

Далее, необходимо вычислить абсолютную погрешность численного интегрирования для обеих формул:

$$E = |K_{ex_int} - K_{func}|, \quad \text{Хорошо, что есть определение в явном виде}$$

где K_{ex_int} - точное значение интеграла для $t_a = \frac{1}{L}$, K_{func} - значение интеграла, полученное через составную формулу Симпсона/трапеций.

Эти вычисления производятся в функции `error_function(a, b, n, func)` и представлены в файлах `basicPart1.ipynb`, `basicPart2.ipynb`.

Далее, просто построим *log-log* график зависимости абсолютной погрешности численного интегрирования от шага интегрирования для обеих формул.

Функция построения графиков - *plotParagraph3()* представлена в Листинге (7). Эта функция одинакова для обоих методов с различием в функции *massive_creation(a, b)* в первом случае и *nodes_creation(a, b)* во втором.

Листинг 7. Реализация функции *plotParagraph3()*

```
1 def plotParagraph3():
2     global dots
3     fig, ax = plt.subplots(figsize=(12, 8))
4
5     n = np.linspace(3, 9999, 9997, dtype=int)
6     a = 1 / dots
7     b = 2
8     massive_creation(a, b)
9     ax.loglog((b - a) / (n - 1), [error_for_function(a, b, n_, composite_simpson) for n_
   in n], 'o', label='simpson', markersize = 2)
10    ax.loglog((b - a) / (n - 1), [error_for_function(a, b, n_, composite_trapezoid) for n_
   in n], 'o', label='trapezoid', markersize = 2)
11
12    ax.grid(True)
13    ax.set_xlabel(r'$h$', fontsize=25)
14    ax.set_ylabel(r'$E$', fontsize=25)
15    ax.legend(fontsize=24)
16
17    plt.tight_layout()
18    plt.xticks(fontsize=18)
19    plt.yticks(fontsize=18)
20    plt.savefig('basicPlot.pdf')
```

Выполненные построения для двух методов численного интегрирования по dx при $L = [10^4; 10^5; 10^6]$:

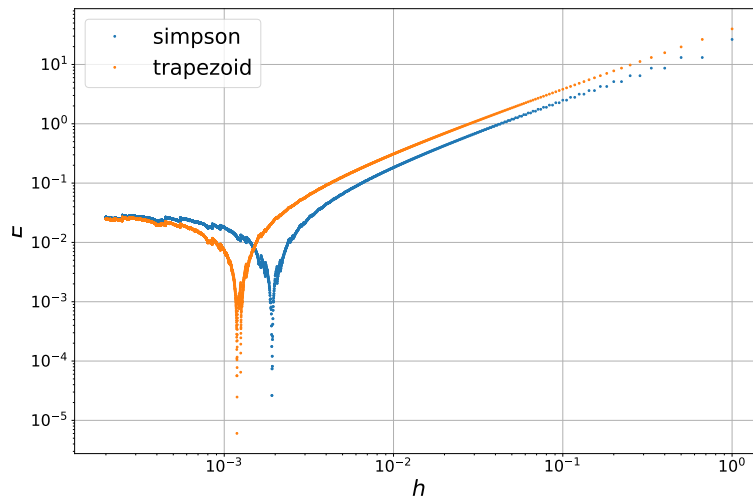


Рис. 1. log-log график зависимости абсолютной погрешности численного интегрирования по dx от шага интегрирования для обеих формул при отступе от 0 на 10^{-4} (первый метод)

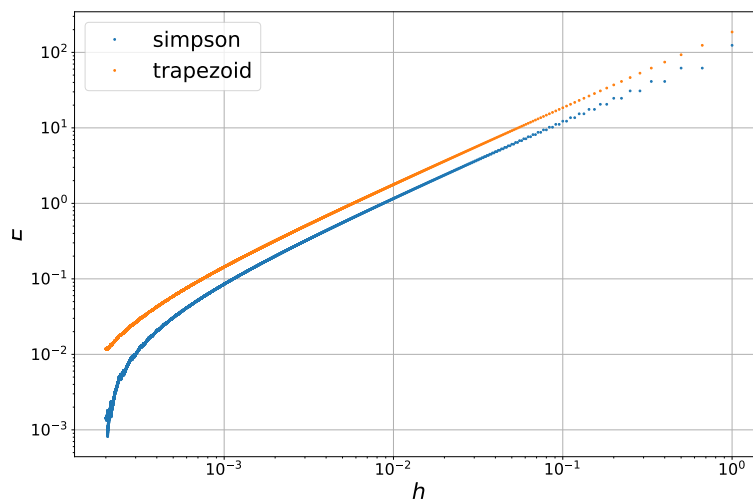


Рис. 2. log-log график зависимости абсолютной погрешности численного интегрирования по dx от шага интегрирования для обеих формул при отступе от 0 на 10^{-5} (первый метод)

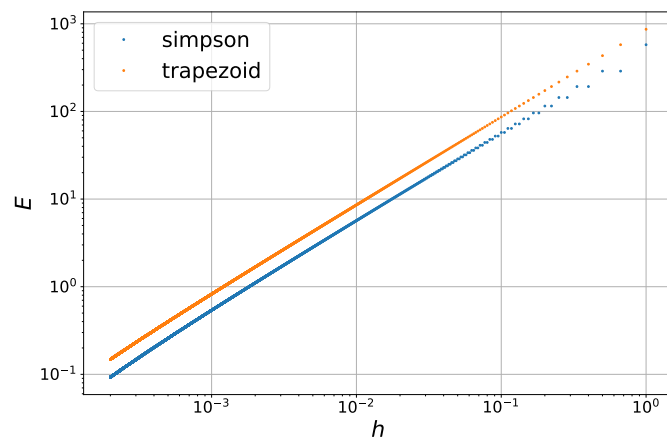


Рис. 3. log-log график зависимости абсолютной погрешности численного интегрирования по dx от шага интегрирования для обеих формул при отступе от 0 на 10^{-6} (первый метод)

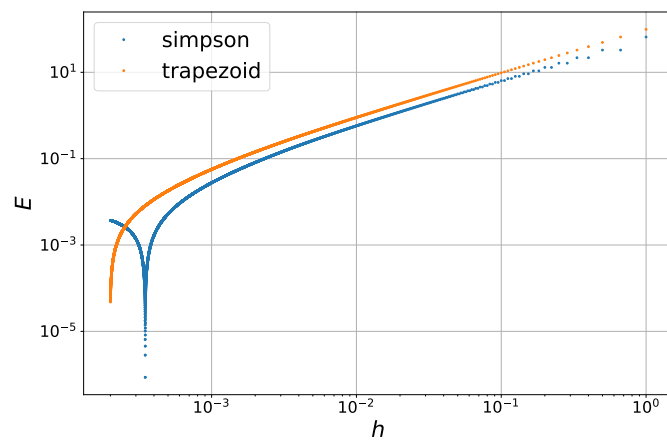


Рис. 4. log-log график зависимости абсолютной погрешности численного интегрирования по dx от шага интегрирования для обеих формул при отступе от 0 на 10^{-4} (второй метод)

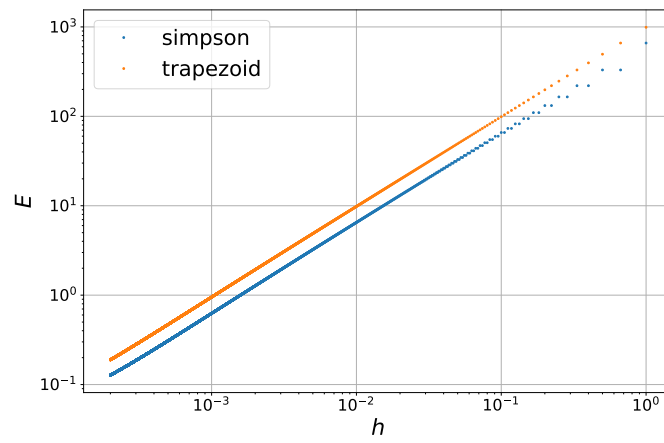


Рис. 5. log-log график зависимости абсолютной погрешности численного интегрирования по dx от шага интегрирования для обеих формул при отступе от 0 на 10^{-5} (второй метод)

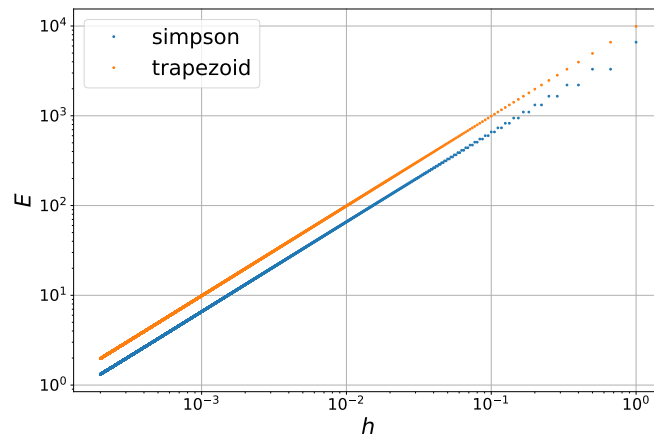


Рис. 6. log-log график зависимости абсолютной погрешности численного интегрирования по dx от шага интегрирования для обеих формул при отступе от 0 на 10^{-6} (второй метод)

Интегрируем по dy (дополнительный подход):

Интегрируем по dy для равномерно распределённых узлов y . Формулы численного интегрирования применяем для интеграла (13). Подынтегральная функция будет иметь следующий вид:

$$f_{\text{under_integral}} = \frac{1}{\sqrt{(C - y) \cdot y}}$$

Её реализация в Листинге (8):

Листинг 8. Реализация функции *under_integral_function(y)*

```
1 def under_integral_function(y):
2     C = 1.03439984
3
4     return 1 / np.sqrt((C - y) * y)
```

Вычисление абсолютной погрешности и построение графика зависимости выполняются с помощью разработанных ранее функций *error_function(a, b, n, func)*, *plotParagraph3()*. Интегрирование производим на отрезке $[\frac{1}{L}; C - \frac{1}{L}]$ и $t_a = \frac{1}{L}$, где $L = [10^4, 10^5, 10^6]$.

Реализация функции *error_function(a, b, n, func)* представлена в файле *basicPart3.ipynb*, а функции *plotParagraph3()* в Листинге (9).

Листинг 9. Реализация функции *plotParagraph3()*

```
1 def plotParagraph3():
2     fig, ax = plt.subplots(figsize=(12, 8))
3
4     C = 1.03439984
5     n = np.linspace(3, 9999, 9997, dtype=int)
6     a = 0.0001
7     b = C - 0.0001
8     ax.loglog((b - a) / (n - 1), [error_for_function(a, b, n_, composite_simpson) for n_
9         in n], 'o', label='simpson', markersize = 2)
10    ax.loglog((b - a) / (n - 1), [error_for_function(a, b, n_, composite_trapezoid) for n_
11        in n], 'o', label='trapezoid', markersize = 2)
12
13    ax.grid(True)
14    ax.set_xlabel(r'$h$', fontsize=16)
15    ax.set_ylabel(r'$E$', fontsize=16)
16    ax.legend(fontsize=16)
17
18    plt.tight_layout()
19    plt.savefig('basicPlot3.pdf')
```

Выполненные построения для численного интегрирования по dy при $L = [10^4; 10^5; 10^6]$:

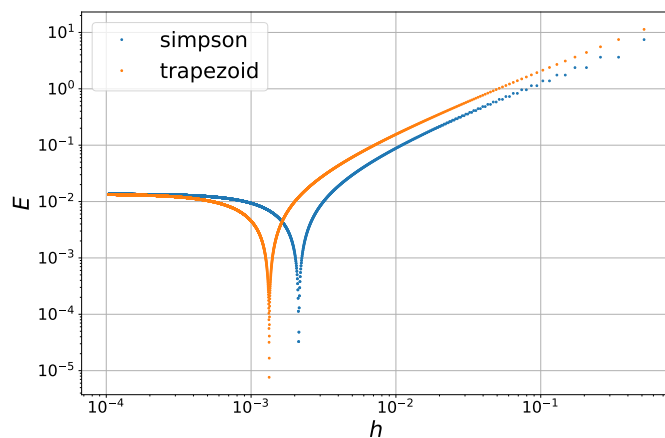


Рис. 7. log-log график зависимости абсолютной погрешности численного интегрирования по dy от шага интегрирования для обеих формул при отступе от 0 на 10^{-4}

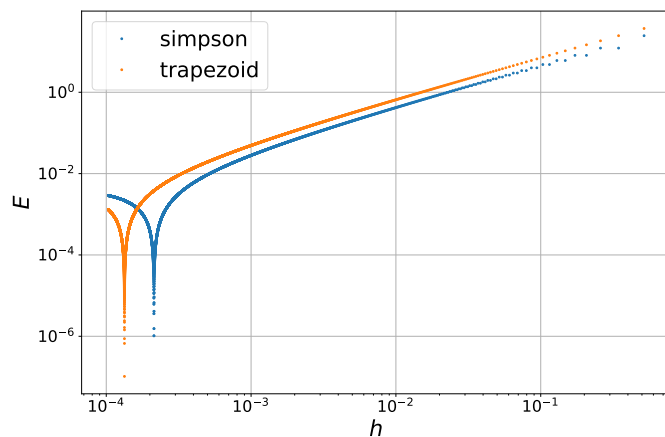


Рис. 8. log-log график зависимости абсолютной погрешности численного интегрирования по dy от шага интегрирования для обеих формул при отступе от 0 на 10^{-5}

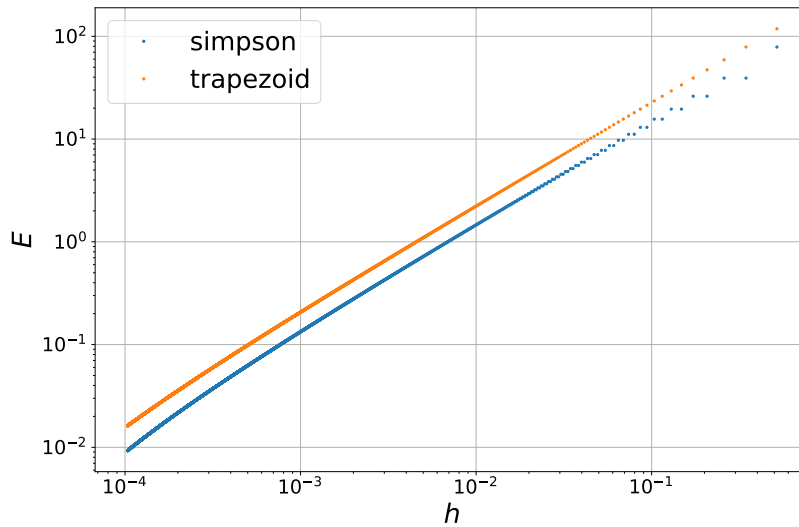


Рис. 9. log-log график зависимости абсолютной погрешности численного интегрирования по dy от шага интегрирования для обеих формул при отступе от 0 на 10^{-6}

Анализ использованных методов и полученных построений:

Далее приводится подробный анализ, это очень хорошо

Для первого метода интегрирования по dx заметим:

- На графике (1) при отступе от 0 на 10^{-4} значения стабильно вычисляются примерно до $\frac{1}{500}$. Далее, с уменьшением шага интегрирования погрешность стремительно уменьшается и достигает значений в 10^{-5} . После этого погрешность для составной формулы Симпсона становится больше чем по формуле трапеций и обе погрешности увеличиваются к значению в $\frac{1}{50}$. **Этот график неудовлетворителен.**
- На графике (2) при отступе от 0 на 10^{-5} такие значительные скачки уже отсутствуют. Погрешность значительно меньше, а значения стабильно вычисляются примерно до $\frac{1}{9500}$. Далее, с уменьшением шага интегрирования график погрешности немного изменяет свою скорость, что кажется неправильным, так как операция численного интегрирования вычислительно-устойчивая, поэтому стабильность вычислений должна быть вплоть до машинного эпсилона. **Этот график неудовлетворителен.**
- На графике (3) при отступе от 0 на 10^{-6} график выглядит неплохо, стабильное линейное изменение погрешности вплоть до 9999 узлов интегрирования. Единственный значительный минус - большая погрешность в $\frac{1}{10}$.

Для второго метода интегрирования по dx заметим:

- График (4) явно похож на график (1). И там, и там отступ от 0 на 10^{-4} , сначала значения вычисляются стабильно, но затем скорость графика резко меняется и погрешность при составной формуле трапеций становится больше, чем при формуле Симпсона. **Этот график неудовлетворителен.**
- На графике (5) при отступе от 0 на 10^{-5} видим идентичную картинку, что и на графике (3) с тем различием, что на втором погрешность чуть меньше.
- График (6) при отступе от 0 на 10^{-6} выглядит так же, стабильный линейный график, но уже с минимальной погрешностью в единицу. **Этот график неудовлетворителен.**

Для численного интегрирования по dy заметим:

- По рассуждениям выше очевидно, что скачкообразные графики не являются приемлемыми, поэтому при рассмотрении графиков (7), (8) делаем вывод, что **они неудовлетворительны.**
- График (9) при отступе от 0 на 10^{-6} линеен на всем отрезке варьирования шага интегрирования. Значение погрешности самое маленькое из полученных в данном исследовании.

Таким образом, выбор происходит между графиками (3), (5), (9). Отталкиваясь от значения погрешности, выбор падает на график (9) с наименьшей погрешностью при $n = 9999$.

В таком случае:

Точное значение, посчитанное через формулу (12) -

$$\sqrt{\frac{2 \cdot C}{g}} \cdot (T - t_a) = \sqrt{\frac{2.06879968}{9.81}} \cdot (1.75418438 - 0.000001) = 0.80556337$$

Значение с выбранного графика и погрешность -

• по составной формуле Симпсона: 0.81486634 и 0.00930298

• по составной формуле трапеций: 0.82178919 и 0.01622583

Здорово, что посчитаны абсолютные погрешности. Это очень наглядно

4. Определение порядка точности формулы по полученному графику.

Чтобы оценить точность выполняемых вычислений, вводится такое понятие оценки точности квадратуры, как **порядок точности**. Это оценка наибольшей степени полинома, который интегрируется квадратурой точно. Порядок точности квадратуры равен $n \in \mathbb{N}$ тогда и только тогда, когда остаточный член равен нулю для всех многочленов степени $\leq n$, и не равен нулю для хотя бы одного многочлена степени $n + 1$.

Хорошо, что определение выделено в тексте

Отлично, что приведены пояснения тому, как мы вычисляем порядок точности по log-log графику

В нашем случае для log-log графика (9) порядок точности можно определить, как тангенс угла наклона графика. Если примем порядок точности за $O(h)$ и построим рядом с нашим графиком прямую, обозначающую единичный порядок точности, то получим:

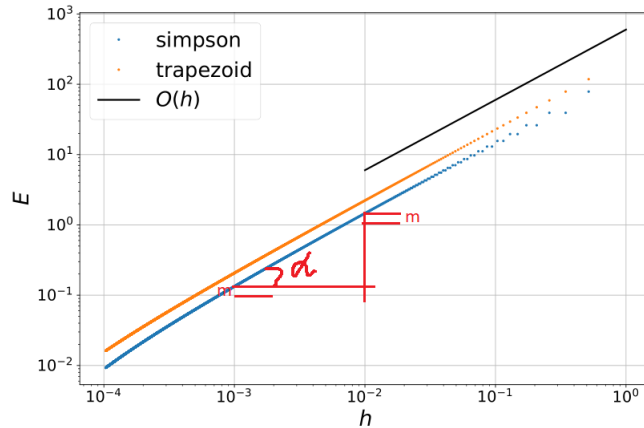


Рис. 10. Оценка порядка точности формул по графику зависимости погрешности численного интегрирования от шага

$$O(h) = \operatorname{tg}(\alpha) = \frac{\lg(10^0) - \lg(10^{-1})}{\lg(10^{-2}) - \lg(10^{-3})} = \frac{0 + 1}{-2 + 3} = 1 \quad \text{Вот это очень наглядная выкладка}$$

Так как графики, полученные формулами Симпсона и трапеций, параллельны, то по полученным графикам делаем вывод, что порядок точности обеих формул равен единице.

5. Сравнение порядка точности, полученный с помощью графика, с аналитическим порядком точности для обеих формул.

В пункте №4 мы получили, что как для составной формулы Симпсона, так и для составной формулы трапеций порядок точности является единичным. Но по остаточным членам из этих формул (3), (5) достаточно ясно, что аналитически эти порядки равны 4 и 2 соответственно.

Такое различие можно объяснить тем, что подынтегральная функция является недостаточно гладкой. Гладкая функция - это такая функция, которая имеет непрерывную производную на множестве определения. Соответственно, составная формула Симпсона по определению требует четвёртую степень гладкости, а составная формула трапеций - вторую.

Да, это верное заключение

Оценим, какую степень гладкости имеет наше подынтегральное выражение:

$$f_{\text{under_integral}} = \frac{1}{\sqrt{(C-y) \cdot y}}$$

Её производная будет иметь следующий вид:

$$f'_{under_integral} = \frac{2 \cdot y - C}{2 \cdot ((C - y) \cdot y)^{\frac{3}{2}}} \quad \text{Хорошо, что вычислена производная}$$

Очевидно, что пределы на границах отрезка интегрирования стремятся к бесконечности, как у самой функции:

$$\begin{aligned} \lim_{y \rightarrow 0+0} \frac{1}{\sqrt{(C - y) \cdot y}} &= \frac{1}{\sqrt{C \cdot (0 + 0)}} = \frac{1}{0 + 0} = +\infty \\ \lim_{y \rightarrow C-0} \frac{1}{\sqrt{(C - y) \cdot y}} &= \frac{1}{\sqrt{(C - (C - 0)) \cdot C}} = \frac{1}{\sqrt{C \cdot (0 + 0)}} = +\infty \end{aligned}$$

Так и у её производной:

$$\begin{aligned} \lim_{y \rightarrow 0+0} \frac{2 \cdot y - C}{2 \cdot ((C - y) \cdot y)^{\frac{3}{2}}} &= \frac{-C}{2 \cdot (C \cdot (0 + 0))^{\frac{3}{2}}} = \frac{-C}{(0 + 0)} = -\infty \\ \lim_{y \rightarrow C-0} \frac{2 \cdot y - C}{2 \cdot ((C - y) \cdot y)^{\frac{3}{2}}} &= \frac{C}{2 \cdot ((0 + 0) \cdot C)^{\frac{3}{2}}} = \frac{C}{(0 + 0)} = +\infty \end{aligned}$$

Таким образом, получаем, что подынтегральная функция и её производная терпят разрыв в точках 0 и C . Поэтому функции не являются непрерывными на всём отрезке от 0 до C , что и является причиной разницы фактических и аналитических порядков точности. Да, здесь все тоже верно. Здорово, что приведены рассуждения о разрыве функции с использованием аппарата математического анализа

6. Анализ существования оптимального шага интегрирования для данной формулы, минимизирующий достижимую погрешность.

Из выводов пункта №3 известно, что оптимального шага существовать не должно, погрешность должна стабильно уменьшаться с увеличением числа узлов интегрирования. Другими словами, вычисления при численном интегрировании стабильны, то есть в отличие от численного дифференцирования операция численного интегрирования вычислительно-устойчивая.

Докажем это на примере составной формулы трапеций (5):

Допустим, что значение $f(x)$ в точке x_i вычисляется с погрешностью округления e_i :

$$f(x_i) = \tilde{f}(x_i) + e_i,$$

где $i = 1, \dots, n$.

Тогда полная погрешность округления составной формулы трапеций может быть оценена, как:

Хорошо было бы упомянуть про неравенство треугольника

$$e(h) = \frac{h}{2} \left[e_1 + 2 \cdot \sum_{i=1}^{n-1} e_i + e_n \right] \leq \frac{h}{e} \cdot \left[|e_1| + 2 \cdot \sum_{i=1}^{n-1} |e_i| + |e_n| \right];$$

Если ограничить погрешность округления машинным эпсилоном ϵ ($|e_i| \leq \epsilon$), то полная погрешность оценивается, как:

$$e(h) \leq \frac{h \cdot \epsilon}{2} \cdot [1 + 2 \cdot (n - 1) + 1] = n \cdot h \cdot \epsilon = (b - a) \cdot \epsilon;$$

Да, это верно

Это означает, что верхняя граница погрешности округления не зависит ни от n , ни от h . То есть независимо от кол-ва подотрезков погрешность интегрирования будет стабильна. Таким образом, заметим, что при интегрировании погрешность уменьшается до тех пор, пока не достигнет сравнимого с машинным эпсилоном значения, после этого уменьшение невозможно и погрешность стабилизируется на уровне машинного эпсилонa. Да, все так

4 Продвинутая часть

1. Преобразование задачи о минимизации функционала (1) к полудискретной форме, используя кусочно-линейную интерполяцию.
2. Преобразование задачи к полностью дискретной форме, используя составную формулу Симпсона.
3. Решение задачи минимизации, используя различные конфигурации дискретизации: с шагом интерполяции и интегрирования от 10^{-3} до 1.
4. Оценка зависимости погрешности решения от шага интерполяции и шага интегрирования, используя log-log графики и линии уровня.

1. Преобразование задачи о минимизации функционала (1) к полудискретной форме, используя кусочно-линейную интерполяцию.

Функционал (1) выглядит следующим образом:

$$\mathcal{F}[y] = \int_0^a \sqrt{\frac{1 + (y'(x))^2}{2gy(x)}} dx;$$

Приведём этот интеграл к полудискретной форме:

$$\sum_{i=0}^{n-1} \int_{x_i}^{x_{i+1}} \sqrt{\frac{1 + (S'_i(x))^2}{2 \cdot g \cdot S_i(x)}} dx, \quad (14)$$

Это правильное представление полудискретной формы

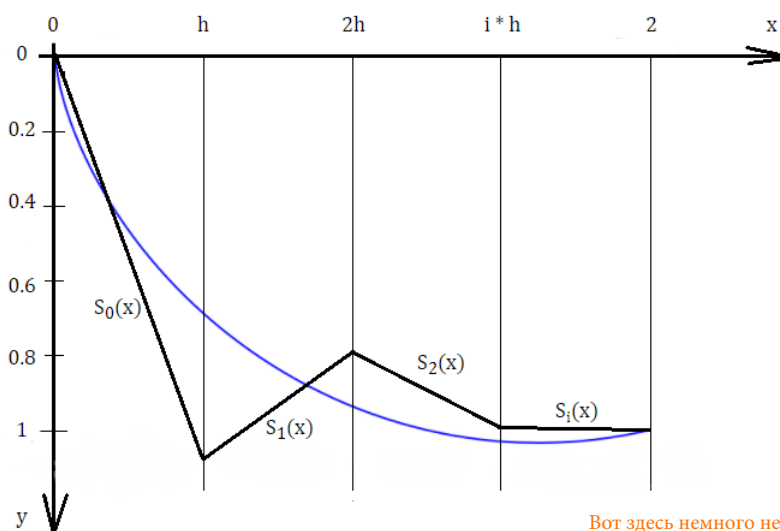
где $S_i(x)$ - некоторый многочлен, приближающий функцию $y(x)$ на каждом подотрезке $[x_i; x_{i+1}]$, $i = 0, \dots, n - 1$, а n - количество узлов интерполяции.

Так как необходимо использовать кусочно-линейную интерполяцию, то пусть $S_i(x)$ имеет следующий вид:

$$S_i(x) = \alpha_i + \beta_i \cdot x, \quad \text{Верно} \quad (15)$$

где $i = 0, \dots, n - 1$.

Тогда на рисунке (11) представлены $S_i(x)$, аппроксимирующие функцию $y(x)$ на отрезке $[0; 2]$.



Вот здесь немного непонятно, если синяя линия это функционал, то тогда не совсем наглядно приводить интерполяцию $y(x)$ с ней на одном графике

Рис. 11. Кусочно-линейная интерполяции функции $y(x)$ для минимизации функционала (1)

Найдем производную от выражения (15):

$$S'_i(x) = \beta_i \quad (16)$$

Подставим (15), (16) в выражение (14):

Да, здесь все выкладки верные

$$H(\alpha, \beta) = \sum_{i=0}^{n-1} \int_{x_i}^{x_{i+1}} \sqrt{\frac{1 + \beta_i^2}{2 \cdot g \cdot (\alpha_i + \beta_i \cdot x)}} dx = \sum_{i=0}^{n-1} \sqrt{\frac{1 + \beta_i^2}{2 \cdot g}} \cdot \int_{x_i}^{x_{i+1}} \frac{dx}{\sqrt{\alpha_i + \beta_i \cdot x}}; \quad (17)$$

Для решения задачи о минимизации функционала в полудискретной форме необходимо найти все коэффициенты α_i и β_i - такие, что функционал (14) будет иметь наименьшее значение. Чтобы их найти необходимо:

1. Найти все частные производные $H(\alpha, \beta)$ по всем α_i и β_i , приравнять их к нулю.
2. Учесть граничные условия, по которым известны ординаты крайних точек отрезка.
3. Учесть условие сопряжения точек между соседними функциями $S_i(x)$.

Граничные условия:

$$S_0(0) = 0; \quad (18)$$

$$S_{n-1}(2) = 1; \quad (19)$$

Условия сопряжения:

$$S_i(x_{i+1}) = S_{i+1}(x_{i+1}), \quad (20)$$

где $i = 0, \dots, n-2$.

Преобразуем выражение (17) для нахождения частных производных:

$$\begin{aligned} H(\alpha, \beta) &= \sum_{i=0}^{n-1} \sqrt{\frac{1+\beta_i^2}{2 \cdot g}} \cdot \int_{x_i}^{x_{i+1}} \frac{dx}{\sqrt{\alpha_i + \beta_i \cdot x}} = \\ &= \sum_{i=0}^{n-1} \sqrt{\frac{1+\beta_i^2}{2 \cdot g}} \cdot \frac{1}{\beta_i} \cdot \int_{x_i}^{x_{i+1}} \frac{d(\alpha_i + \beta_i \cdot x)}{\sqrt{\alpha_i + \beta_i \cdot x}} = \\ &= \sum_{i=0}^{n-1} \sqrt{\frac{1+\beta_i^2}{2 \cdot g}} \cdot \frac{2}{\beta_i} \cdot \sqrt{\alpha_i + \beta_i \cdot x} \Big|_{x_i}^{x_{i+1}} = \\ &= \sum_{i=0}^{n-1} \sqrt{\frac{1+\beta_i^2}{2 \cdot g}} \cdot \frac{2}{\beta_i} \cdot (\sqrt{\alpha_i + \beta_i \cdot x_{i+1}} - \sqrt{\alpha_i + \beta_i \cdot x_i}), \end{aligned} \quad (21)$$

Найдём частные производные $H(\alpha, \beta)$ по α_i :

$$\begin{aligned} \frac{\partial H(\alpha, \beta)}{\partial \alpha_i} &= \sqrt{\frac{1+\beta_i^2}{2 \cdot g}} \cdot \frac{2}{\beta_i} \cdot \frac{\partial(\sqrt{\alpha_i + \beta_i \cdot x_{i+1}} - \sqrt{\alpha_i + \beta_i \cdot x_i})}{\partial \alpha_i} = \\ &= \sqrt{\frac{1+\beta_i^2}{2 \cdot g}} \cdot \frac{2}{\beta_i} \cdot \left(\frac{1}{2 \cdot \sqrt{\alpha_i + \beta_i \cdot x_{i+1}}} - \frac{1}{2 \cdot \sqrt{\alpha_i + \beta_i \cdot x_i}} \right) = \\ &= \sqrt{\frac{1+\beta_i^2}{2 \cdot g}} \cdot \frac{1}{\beta_i} \cdot \left(\frac{\sqrt{\alpha_i + \beta_i \cdot x_i} - \sqrt{\alpha_i + \beta_i \cdot x_{i+1}}}{\sqrt{\alpha_i + \beta_i \cdot x_{i+1}} \cdot \sqrt{\alpha_i + \beta_i \cdot x_i}} \right), \end{aligned} \quad (22)$$

где $i = 0, \dots, n-1$

Найдём частные производные $H(\alpha, \beta)$ по β_i :

$$\begin{aligned} \frac{\partial H(\alpha, \beta)}{\partial \beta_i} &= \frac{\partial \left(\sqrt{\frac{1+\beta_i^2}{2 \cdot g}} \cdot \frac{2}{\beta_i} \right)}{\partial \beta_i} \cdot (\sqrt{\alpha_i + \beta_i \cdot x_{i+1}} - \sqrt{\alpha_i + \beta_i \cdot x_i}) + \\ &+ \sqrt{\frac{1+\beta_i^2}{2 \cdot g}} \cdot \frac{2}{\beta_i} \cdot \frac{\partial(\sqrt{\alpha_i + \beta_i \cdot x_{i+1}} - \sqrt{\alpha_i + \beta_i \cdot x_i})}{\partial \beta_i} = \\ &= \frac{2}{\sqrt{2 \cdot g}} \cdot \left(\frac{1}{\sqrt{1+\beta_i^2}} + \sqrt{1+\beta_i^2} \cdot \ln(\beta_i) \right) \cdot (\sqrt{\alpha_i + \beta_i \cdot x_{i+1}} - \sqrt{\alpha_i + \beta_i \cdot x_i}) + \\ &+ \sqrt{\frac{1+\beta_i^2}{2 \cdot g}} \cdot \frac{2}{\beta_i} \cdot \frac{\sqrt{\alpha_i + \beta_i \cdot x_i} \cdot x_{i+1} - \sqrt{\alpha_i + \beta_i \cdot x_{i+1}} \cdot x_i}{2 \cdot \sqrt{\alpha_i + \beta_i \cdot x_{i+1}} \cdot \sqrt{\alpha_i + \beta_i \cdot x_i}}, \end{aligned} \quad (23)$$

где $i = 0, \dots, n-1$

Выражения выше (18), (19), (20), (22), (23) собираются в систему:

$$\begin{cases} S_0(0) = 0; \\ S_{n-1}(2) = 1; \\ S_i(x_{i+1}) = S_{i+1}(x_{i+1}); & i = 0, \dots, n-2 \\ \frac{\partial H(\alpha, \beta)}{\partial \alpha_i} = \sqrt{\frac{1 + \beta_i^2}{2 \cdot g}} \cdot \frac{1}{\beta_i} \cdot \left(\frac{\sqrt{\alpha_i + \beta_i \cdot x_i} - \sqrt{\alpha_i + \beta_i \cdot x_{i+1}}}{\sqrt{\alpha_i + \beta_i \cdot x_{i+1}} \cdot \sqrt{\alpha_i + \beta_i \cdot x_i}} \right) = 0; & i = 0, \dots, n-1 \\ \frac{\partial H(\alpha, \beta)}{\partial \beta_i} = \frac{2}{\sqrt{2 \cdot g}} \cdot \left(\frac{1}{\sqrt{1 + \beta_i^2}} + \sqrt{1 + \beta_i^2} \cdot \ln(\beta_i) \right) \cdot (\sqrt{\alpha_i + \beta_i \cdot x_{i+1}} - \sqrt{\alpha_i + \beta_i \cdot x_i}) + \\ + \sqrt{\frac{1 + \beta_i^2}{2 \cdot g}} \cdot \frac{2}{\beta_i} \cdot \frac{\sqrt{\alpha_i + \beta_i \cdot x_i} \cdot x_{i+1} - \sqrt{\alpha_i + \beta_i \cdot x_{i+1}} \cdot x_i}{2 \cdot \sqrt{\alpha_i + \beta_i \cdot x_{i+1}} \cdot \sqrt{\alpha_i + \beta_i \cdot x_i}} = 0; & i = 0, \dots, n-1 \end{cases}$$

Очевидно, решив данную систему уравнений, мы найдём все коэффициенты, параметры интерполяции. Попробуем решить:

$$\begin{aligned} \frac{\partial H(\alpha, \beta)}{\partial \alpha_i} &= \sqrt{\frac{1 + \beta_i^2}{2 \cdot g}} \cdot \frac{1}{\beta_i} \cdot \left(\frac{\sqrt{\alpha_i + \beta_i \cdot x_i} - \sqrt{\alpha_i + \beta_i \cdot x_{i+1}}}{\sqrt{\alpha_i + \beta_i \cdot x_{i+1}} \cdot \sqrt{\alpha_i + \beta_i \cdot x_i}} \right) = 0 \\ \sqrt{\alpha_i + \beta_i \cdot x_i} - \sqrt{\alpha_i + \beta_i \cdot x_{i+1}} &= 0 \\ \sqrt{\alpha_i + \beta_i \cdot x_i} &= \sqrt{\alpha_i + \beta_i \cdot x_{i+1}} \\ \alpha_i + \beta_i \cdot x_i &= \alpha_i + \beta_i \cdot x_{i+1} \\ x_i &= x_{i+1} - \text{ЛОЖЬ} \end{aligned}$$

Хорошо, что выше приведены очень подробные математические выкладки

где $i = 0, \dots, n-1$

Делаем вывод, что вся система уравнений не имеет решение, используя такой метод минимизации.

Хорошо, что этому приведено обоснование

2. Преобразование задачи к полностью дискретной форме, используя составную формулу Симпсона.

Допустим, пункт №1 был выполнен и найдены параметры кусочно-линейной интерполяции α_i, β_i , где $i = 0, \dots, n_1 - 1$, n_1 - количество узлов интерполяции, а $h_1 = \frac{2}{n_1 - 1}$ - шаг интерполяции.

Тогда к каждому интегралу в следующем выражении можно применить составную формулу Симпсона:

$$H(\alpha, \beta) = \sum_{i=0}^{n_1-1} \sqrt{\frac{1 + \beta_i^2}{2 \cdot g}} \cdot \int_{x_i}^{x_{i+1}} \frac{dx}{\sqrt{\alpha_i + \beta_i \cdot x}}; \quad (24)$$

Левая часть формулы Симпсона будет иметь следующий вид:

$$\int_a^b f(x) dx \approx \frac{h_2}{3} \cdot \left[f(x_1) + 2 \cdot \sum_{j=1}^{(n_2-3)/2} f(x_{2i+1}) + 4 \cdot \sum_{j=1}^{(n_2-1)/2} f(x_{2i}) + f(x_{n_2}) \right]; \quad (25)$$

где $f(x) = \frac{1}{\sqrt{\alpha_i + \beta_i \cdot x}}$, $a = x_i$, $b = x_{i+1}$, $h_2 = \frac{x_{i+1} - x_i}{n_2 - 1}$ - шаг интегрирования, а n_2 - **КОЛ-ВО** узлов интегрирования. Лучше писать полностью

Её реализация в Листинге (1):

Таким образом, по завершении пункта №2 задача минимизации будет приведена полностью к дискретной форме.

Да, все рассуждения показывают алгоритм, как можно получить полностью дискретную форму

5 Заключение

1. По завершении базовой части:

- были изучены методы численного интегрирования квадратурой, на языке программирования Python реализовали два метода - интегрирование при помощи составной формулы Симпсона и составной формулы трапеций
- была рассмотрена задача о брахистохроне, проанализировали её решение и привели различные методы по численному интегрированию функционала, который является полным временем движения точки по кривой наискорейшего спуска
- для одного из трёх методов интегрирования исследовали порядок точности формулы численного интегрирования на практике и сравнили с теорией
- было опровергнуто существование оптимального шага интегрирования для данной формулы, минимизирующий достижимую погрешность, при том условии, что интегрирование было выполнено для корректных начальных условий Стоило все же здесь указать, что оптимальный шаг имеет некую зависимость от машинного эпсилон

2. По завершении продвинутой части:



- была преобразована задача о минимизации функционала к полудискретной форме, были приведены условия и система уравнений, которая приводит к решению данной задачи, а также была предпринята попытка поиска такого решения, но аналитически данное уравнение не было решено или вовсе не имеет решения
- с использованием метода численного интегрирования составной формулой Симпсона из базовой части задача о минимизации была полностью приведена к дискретной форме

Список использованных источников

1. Першин А.Ю. Лекции по курсу «Вычислительная математика». Москва, 2018-2021. С. 140.

Выходные данные

Очиров Б. Ю.. Отчет о выполнении лабораторной работы по дисциплине «Вычислительная математика». [Электронный ресурс] — Москва: 2021. — 27 с. URL: <https://sa2systems.ru:88> (система контроля версий кафедры РК6)

Постановка:  ассистент кафедры РК-6, PhD А.Ю. Першин
Решение и вёрстка:  студент группы РК6-55Б, Очиров Б. Ю.

2021, осенний семестр