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:	Караф Сармат Майк
Группа:	РК6-52Б
Тип задания:	лабораторная работа
Тема:	Использование аппроксимаций для численной оптимизации

Студент	_______________ подпись, дата	Караф С.М. Фамилия, И.О.
Преподаватель	_______________ подпись, дата	Соколов А. П. Фамилия, И.О.

Москва, 2021

Содержание

Использование аппроксимаций для численной оптимизации

1 Задание

Методы аппроксимации, такие как интерполяция и численное интегрирование, частоиспользуются как составные блоки других, более сложных численных методов. Вданной лабораторной работе мы рассмотрим одну из старейших задач вариационногоисчисления: задачу о брахистохроне, т.е. задачу о кривой наискорейшего спуска. Онасостоит в нахождении такой кривой, по которой материальная точка из точки (x,y) = (0,0) достигнет точки (x,y) = (a, y_b) под действием силы тяжести за наименьшеевремя (здесь и далее ось y направлена вниз). Решением этой задачи является такаякривая y(x), которая минимизирует следующий функционал, являющийся полнымвременем движения материальной точки:

$$F[y] = \int_a^b \sqrt{\frac{1 + (y'(x))^2}{2gy(x)}} dx, \tag{1}$$

где g обозначает ускорение свободного падение, и $y(x) = dy/dx$. Эта задача имеетаналитическое решение, которым является параметрически заданная циклоида:

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

где t $\in [0; T]$ и C,T являются константами, значения которых находятся из граничного условия. В базовой части требуется воспользоваться численным интегрированиемдля нахождения полного времени движения материальной точки по кривой наискорейшего спуска. В продвинутой части требуется разработать метод для нахожденияаппроксимации этой кривой. Здесь и далее принимается b = 2 и y_b= 1. Константыциклоиды для этого граничного условия равны C = 1.03439984, T = 1.75418438

Базовая часть:

1. Написать функцию composite_simpson(a, b, n, f) численного интегрированияфункции f на интервале [a; b] по n узлам с помощью составной формулы Симпсона.

2. Написать функцию composite_trapezoid(a, b, n, f) численного интегрированияфункции 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-log график зависимостиабсолютной погрешности численного интегрирования от шага интегрирования дляобоих формул.

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

5. Для обоих формул сравните порядок, полученный с помощью графика, с аналитическим порядком точности.

6. Существует ли оптимальный шаг интегрирования для данной формулы, минизимирующий 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-log графики и линии уровня, оценить зависимость погрешности решения от шага интерполяции и шага интегрирования.

2 Цель выполнения лабораторной работы

Цель выполнения лабораторной работы – Найти полное время движения материальной точки по кривой наискорейшего спуска, воспользовавшись численным интегрированием при помощи составных формул Сипсона и трапеции, проанализировать зависимость величины абсолютной погрешности от шага интегрирование для реализованных методов.

3 Выполненные задачи

Базовая часть:

1. Разработана функция *composite_simpson(a, b, n, f)* численного интегрирования функции f на интервале [a; b] по n узлам с помощью составной формулы Симпсона.

2. Разработана функция *composite_trapezoid(a, b, n, f)* численного интегрирования функции f на интервале [a; b] по n узлам с помощью составной формулы трапеций.

3. Построен график зависимости абсолютной погрешности численного интегрирования от шага интегрирования для обоих формул, расчитав интеграл (1) для функции y(x), соответствующей кривой наискорейшего спуска.

4. Определить порядок точности формулы по полученному графику .

5. Сравнить порядок, полученный с помощью графика, с аналитическим порядком точности.

6. Определить оптимальный шаг интегрирования для формул минизимирующий полученную погрешность.

4 Базовая часть

4.1 Разрабтка функции для целочисленного интергрирования с помощью составной формулы Симпсона.

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

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

Для нахождение числного значения интергрaла функции (3) с помощью составной формулы Симпсона разработаня функция *composite_simpson(a, b, n, f)* на языке Python.

Функция *composite_simpson(a, b, n, f)*, реализовывающая эти вычисления, представлена в файле Lab_2_Base.py и в Листинге 1.

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

```
def composite_simpson(a, b, n, f):
    if n % 2 != 0:
        n = n + 1
    x = np.linspace(a, b, n + 1)
    h = (b − a) / n
    sum_odd = np.sum([f(x_) for x_ in x[2:−1:2]])
    sum_even = np.sum([f(x_) for x_ in x[1::2]])
    return h / 3. ∗ (
            f(x[0]) + 2. ∗ sum_odd + 4. ∗ sum_even + f(x[−1]))
```

Входные данные функции: a - нижний предел интегрирования, b - верхний предел интегрирования, n - число узлов, в которых должна быть вычислена подынтегральная функция, h - шаг интегрирование, f - интегрируемая функция (1).

Выходные данные: численное значение интеграл при помощи составной формулы Симпсона (3).

4.2 Разрабтка функции для целочисленного интергрирования с помощью составной формулы трапеции.

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

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

Для нахождение числного значения интергрaла функции (4) с помощью составной формулы трапеции разработаня функция *composite_trapezoid(a, b, n, f)* на языке Python.

Функция *composite_trapezoid(a, b, n, f)*, реализовывающая эти вычисления, представлена в файле Lab_2_Base.py и в Листинге 2.

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

```
def composite_trapezoid(a, b, n, f):
    x = np.linspace(a, b, n + 1)
    h = (b - a) / n
    sum_trap = np.sum([f(x_) for x_ in x[1:-1:1]])
    return h / 2. * (f(x[0]) + 2 * sum_trap + f(x[-1]))
```

Входные данные функции: a - нижний предел интегрирования, b - верхний предел интегрирования, n - число узлов, в которых должна быть вычислена подынтегральная функция, h - шаг интегрирование, f - интегрируемая функция (1).

Выходные данные: численное значение интеграл при помощи составной формулы трапеции (4).3,4)

4.3 Построение графика зависимости абсолютной погрешности численного интегрирования от шага интегрирования для обоих формул

Для построения графика зависимости абсолютной погрешности численного интегрирования от шага интегрирования (1) для обоих методов, аппроксимируем кубическим и производным кубического сплайна $y(x)$,$y'(x)$ (Листинге 3 , 4) при заданных коэффициентах C = 1.03439984 и T = 1.75418438, при $t \in [0, T]$. Для этого выполним последовательность действий, описанные в лабораторной работе № 1 по вычислительной математике[1]

Функция *qubic_spline(x, qs_coeff, x_nodes, y_nodes)*, реализовывающая вычисления $y(x)$, представлена в файле Lab_2_Base.py и в Листинге 3.

Листинг 3. Реализация функции *qubic_spline(x, qs_coeff, x_nodes, y_nodes)*

```
def qubic_spline(x, qs_coeff, x_nodes, y_nodes):
    ind = binary_search(x_nodes, x)
    if ind == 1e-21:
        return ind
    return y_nodes[ind] + (qs_coeff[0][ind] * (x - x_nodes[ind])) + (qs_coeff[1][ind] * ((x
        - x_nodes[ind]) ** 2)) + (qs_coeff[2][ind] * (x - x_nodes[ind]) ** 3)
```

Функция *d_qubic_spline(x, qs_coeff, x_nodes)*, реализовывающая вычисления $y'(x)$, представлена в файле Lab_2_Base.py и в Листинге 4.

[1]Ссылка на первую лаборотоную работу https://sa2systems.ru:88/edu/educmm/Karaf_S.M

Листинг 4. Реализация функции *d_ qubic_ spline(x, qs_ coeff, x_ nodes)*

```
def d_qubic_spline(x, qs_coeff, x_nodes):
    ind = binary_search(x_nodes, x)
    if ind == 1e−21:
        return 0
    return qs_coeff[0][ind] + (2 ∗ qs_coeff[1][ind] ∗ (x − x_nodes[ind]))
           + (3 ∗ qs_coeff[2][ind] ∗ ((x − x_nodes[ind]) ∗∗ 2))
```

Далее становится возможным выполнить целочисленное интегрирование. Так как в нижнем пределе интегрирования подынтегральное выражение (1) стремится к бесконечности, необходимо выбрать близкую, но не равную нулю значение. В данной работе было выбрано значение $a = 0.01$.

Разработана функция *f_ y(x)*, реализовывающая вычисления подынтегрального выражения, представлена в файле Lab_2_Base.py и в Листинге 5.

Листинг 5. Реализация функции *f_ x(x)*

```
def f_y(x):
    d_qs = d_qubic_spline(x, qs_coeff, x_nodes)
    qs = qubic_spline(x, qs_coeff, x_nodes, y_nodes)
    return np.sqrt((1 + (d_qs ∗∗ 2)) / (2 ∗ 9.8 ∗ qs))
```

Полученные значения подынтегрального выражения подставим в составные формулы (3), (4) и получим численное решение интеграла. Но, необходимо учесть, что данное решение является приближенным. Чтобы оценить погрешность решения построим график зависимости погрешности численного интегрирования от шага интегрирования. Значения абсолютной погрешности было рассчитано функциями composite_simpson и composite_trapezoid для 20000 узлов, что является приближенным к действительному значению.Также, были разработаны функции error_cs(a, b, n, f_x) и error_ct(a, b, n, f_x), которые вычисляют абсолютную погрешность (Листинг 6) .Шаг интегрирования задается количеством узлов $n \in [3, 9999]$ и пределами интегрирования a = 0.01 b = 2.(Листинг 7)

Листинг 6. Реализация функций *error_ cs(a, b, n, f_ x) и error_ ct(a, b, n, f_ x)*

```
def error_cs(a, b, n, f_x):
    return np.abs(exact_value_s − composite_simpson(a, b, n, f_x))

def error_ct(a, b, n, f_x):
    return np.abs(exact_value_t − composite_trapezoid(a, b, n, f_x))
```

Алгоритм построения log-log графиков зависимости абсолютной погрешности от шага интегрирования для обеих методов , представлена в файле Lab_2_Base.py и в Листинге 7.

Листинг 7. Реализация алоритма построение log-log графиков

```
for n in range(3, 9999):
    Sims.append(error_cs(a, b, n, f_x))
    Trap.append(error_ct(a, b, n, f_x))
    h1.append((b - a) / n)
h_scaling = np.logspace(-4, -2, 999)
plt.loglog(h1, Sims, 'o', markersize=1, label=r'simpson', )
plt.loglog(h1, Trap, 'o', markersize=1, label=r'trapezoid')
plt.legend()
plt.grid()
plt.show()
```

В результате получиться log-log график зависимости абсолютной погрешности от шага интегрирования для обеих методов (Рис.1)

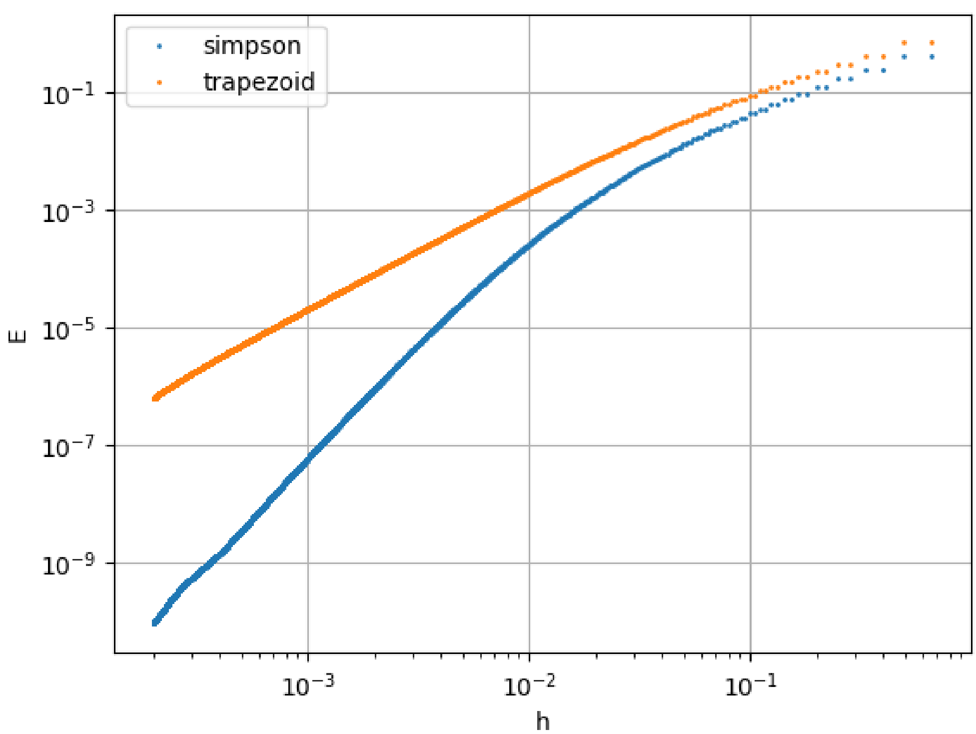

Рис. 1. log-log график зависимости абсолютной погрешности отшага интегрирования для обеих методов при $n[0, 9999]$

4.4 Определние порядка точности формулы по полученному графику

По полученному графику (Рис.1) возможно определение порядка точности.

Остаточный член составной формулы Симпсона (3) пропорционален $O(h^4)$, значит что аналитический порядок точности тоже пропорционален $O(h^4)$.

Остаточный член составной формулы трапеции (4) пропорционален $O(h^2)$. Значит, и аналитический порядок точности пропорционален $O(h^2)$.

Остаточный член составной формулы Симпсона и трапеции имеют следующую форму:

$$-\frac{(b-a)h^4}{180}f^{(4)}(\xi), \tag{5}$$

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

$$-\frac{(b-a)h^2}{12}f''(\xi), \tag{6}$$

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

Выведем порядки точности $O(h)$, $O(h^2)$ и $O(h^4)$, для определения порядка точности погрешности, чтобы по наклону графика погрешности к оси h можно было определить приблизительный порядок точности.(Рис.2)

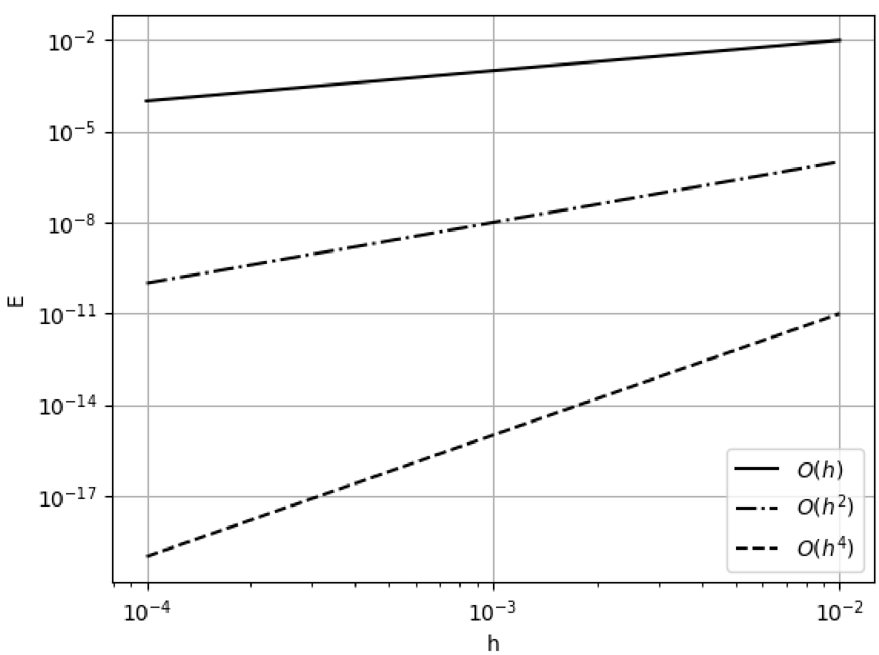

Рис. 2. порядки точности $O(h)$, $O(h^2)$ и $O(h^4)$

4.5 Сравнение порядок полученного с помощью графика, с аналитическим порядком точности

После определения порядка точности, можно отобразить на графике (Рис.1) зависимость погрешности от шага интегрирования обоими методами (3,4) и аналитическую точность (5,6)(Рис 3)

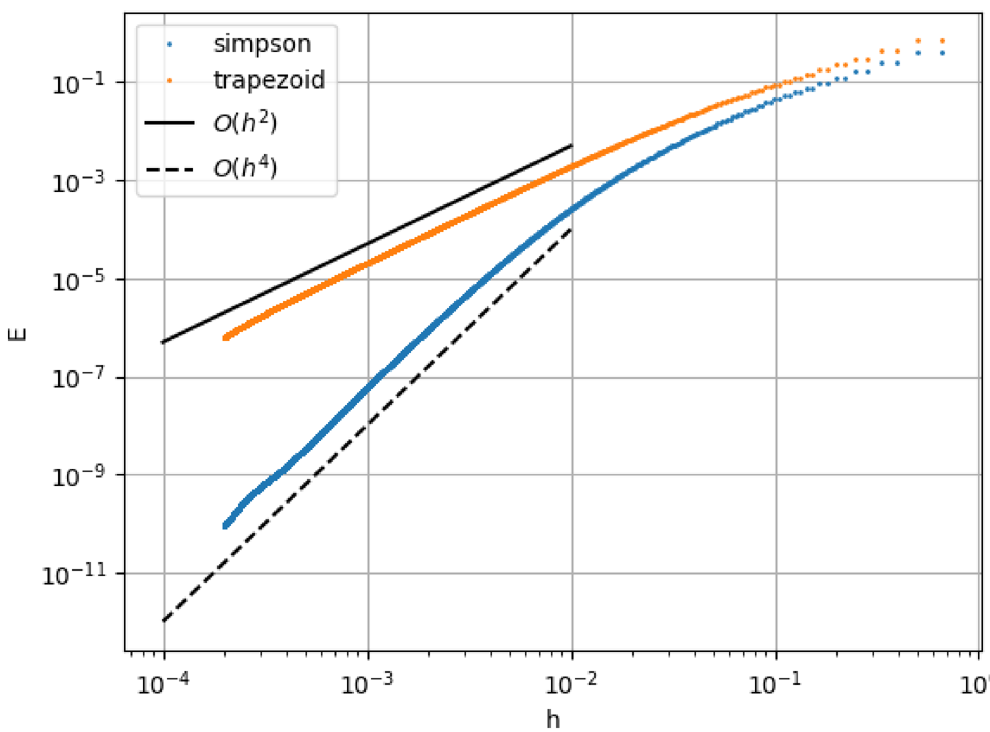

Рис. 3. log-log график зависимости абсолютной погрешности от шага интегрирования для обеих методов с выводом аналитических порядков точности

На графике (Рис. 3) видно, что численное значение погрешностей методов Симпсона и трапеции сильно приближенной к их аналитическому порядку (3,4)

4.6 Определение оптимального шага интегрирования для формул минизимирующий полученную погрешность

После расчётов интеграла (1) формулами Симпсона и трапеции (3, 4) и найденной погрешности с остаточным членом для каждого случая, получается график (Рис .3). Так как погрешность расчётов интеграла не достигла машинного эпсилона, определить оптимальный шаг интегрирование нельзя однозначно

5 Заключение

По итогу выполненной работы были реализованные функции численного интегрирования составных формул Симпсона и трапеции, были вычислены абсолютные погрешности составных формул. Рассмотрена задача о кривой скорейшего спуска. Разработан метод нахождения аппроксимации, для решения задачи минимизации. Построены графики зависимости абсолютной погрешности от выбранных методов, так же были сделаны выводы по порядку точности и наличие оптимального шага

Список использованных источников

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

2. Документация Python 3.9 [Электронный ресурс] [Офиц. сайт]. 2021. (дата обращения 27.10.2021).

Выходные данные

Караф С.М.. Отчет о выполнении лабораторной работы по дисциплине «Вычислительная математика». [Электронный ресурс] — Москва: 2021. — 12 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 РК6-52Б, Караф С.М.*

2021, осенний семестр