БЕЛОРУССКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ
Факультет прикладной математики и информатики
Кафедра вычислительной математики

Лабораторная работа № 2
Численное интегрирование
Вариант № 14

Выполнил:
Кражевский Алексей Игоревич,
3 курс 13 группа (ТП)
Преподаватель:
Будник Анатолий Михайлович

Минск, 2022

Часть 1. Определение шага интегрирования

Постановка задачи

Дан интеграл $I=\int_a^b f(x)dx$, где $a,b,f(x)$ берутся из условия лабораторной работы 1.

Необходимо пользуясь выражением для погрешности интерполирования определить шаг h в СКФ, которая обеспечивает определение интеграла с точностью $\xi=10^{-5}$.

Интегрирование производится согласно варианта 14 (методом трапеций).

Исходные данные:

- $\alpha_j=0{,}1+0{,}05\cdot j, j=14$. Получаем $\alpha_j=0{,}8$
- $[a,b]=[\alpha_j,1+\alpha_j]$. Получаем $[a,b]=[0.8,1.8]$
- $f(x)=\alpha_j e^x+(1-\alpha_j)\sin(x)$. Получаем $f(x)=0.8e^x+0.2\sin(x)$

Алгоритм решения

Квадратурная формула трапеций:

$$\sum_{i=0}^{N}\frac{h}{2}(f_{i-1}+f_i)=h(\frac{1}{2}(f_0+f_N)+\sum_{i=1}^{N}f_i)$$

Остаток формулы трапеций:

$$R_n(f)=\frac{-(b-a)^3}{12N^2}f''(\eta)=O(h^2)$$

Оценка погрешности сверху:

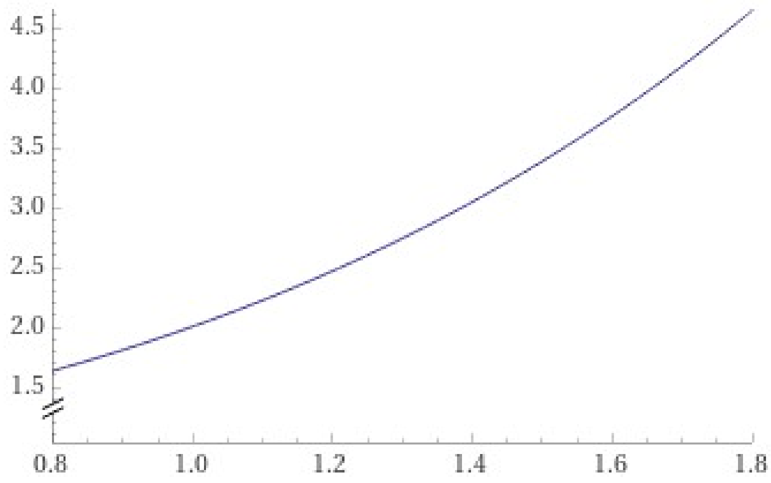
$$|R|=\frac{-(b-a)^3}{12N^2}\cdot M$$

Где $M=max|f''(\eta)|, \eta\in[a,b]$

Формула для нахождения шага:

$$h=\frac{b-a}{N}$$

График второй производной:



Откуда видно, что максимальное значение второй производной достигается в точке 1.8.

Код решения

Функция для интегрирования:

```
def func(x: float) -> float:
    return 0.8 * np.exp(x) + 0.2 * np.sin(x)
```

Вторая производная функции:

```
def second_derivative(x: float) -> float:
    return 0.8 * np.exp(x) - 0.2 * np.sin(x)
```

Функция для подсчета интеграла методом трапеций:

```
def trap_kf(h, n, list_x):
    ans = 0
    for i in range(1, n):
        ans += func(list_x[i])
    ans += (0.5 * (func(list_x[0]) + func(list_x[n])))
    return ans * h
```

Функция для подсчета остатка:

```
def find_R(N: int) -> float:
    eta = 1.8
    return np.abs(((-np.power(b-a, 3)) * second_derivative(eta)) / (12 * (np.power(N, 2))))
```

Функция для нахождения N:

```
def find_N(eps):
    N = 1
    cur = find_R(N)
    while cur >= eps:
        N += 1
        cur = find_R(N)
    return N
```

Функция для нахождения шага:

```
def get_h(N):
    return (b - a) / N
```

Вывод программы

```
N = 197
h = 0.005076142131979695
R_n(x) <= 9.973950297943602e-06
Integration result: 3.2440731618714587
Real integral res: 3.244066989544434
Real error: 6.172327024600577e-06
```

Выводы

Мы вычислили значение N такое, что остаток $R_n(f) \leq 10^{-5}$. Получили, что при разбиении отрезка $[a,b]$ на $N=197$ отрезков мы получаем остаток $|R| \leq 9.9735 \cdot 10^{-6}$, что удовлетворяет постановке задачи. Найденный шаг получился $h=0.005076$.

Часть 2. Вычисление интеграла с помощью КФ Гаусса

Постановка задачи

Для вычисления интеграла применим квадратурную формулу Гаусса при заданном значении n. Оценим погрешность интегрирования через $R_n(f)$.

Исходные данные:

- Согласно варианта 14, $n=3$
- $\alpha_j = 0{,}1 + 0{,}05 \cdot j, j=14$. Получаем $\alpha_j = 0{,}8$
- $[a,b] = [\alpha_j, 1+\alpha_j]$. Получаем $[a,b] = [0.8, 1.8]$
- $f(x) = \alpha_j e^x + (1-\alpha_j)\sin(x)$. Получаем $f(x) = 0.8e^x + 0.2\sin(x)$

Алгоритм решения

Формула Гаусса:

$$\int_{-1}^{1} f(x_k)\,dx \approx \sum_{k=0}^{n} A_k f(x_k)$$

Формула Гаусса на отрезке $[a,b]$:

$$\int_a^b f(x)dx = \frac{b-a}{2}\int_{-1}^{1}\underbrace{f\Big(\frac{a+b}{2}+\frac{b-a}{2}t\Big)}_{g(t)}dt = \frac{b-a}{2}\int_{-1}^{1} g(t)dt \approx$$

$$\approx \frac{b-a}{2}\sum_{i=0}^{n} A_i g(x_i) = \frac{b-a}{2}\sum_{i=0}^{n} A_i f\Big(\frac{a+b}{2}+\frac{b-a}{2}x_i\Big).$$

Формула для нахождения A_k:

$$A_k=\frac{2}{(1-x_k^2)(P'_{n+1}(x_k))^2}$$

Формула многочлена Лежандра:

$$P_n(x)=\left(\frac{1}{2^n n!}\right)\cdot\left(\frac{d^n(x^2-1)^n}{dx^n}\right), n=\overline{0,1,\ldots}$$

В частных случаях получаем:

$$P_0(x)=1$$

$$P_1(x)=\frac{1}{2}\cdot 2x=x$$

Формула для оценки погрешности на отрезке $[-1,1]$:

$$R_n(f)=\frac{2^{2n+3}}{(2n+3)(2n+2)!}\left[\frac{(n+1)!^2}{(2n+2)!}\right]^2\cdot f^{(2n+2)}(\eta)$$

Отобразим эту формулу на отрезок $[a,b]$

$$R_n(f)=\left(\frac{(b-a)}{2}\right)^{2n+3} R_n^{[-1,1]}(f)$$

Для заданного значения $n=3$ получаем:

$$P_4(x)=\frac{1}{2^4 4!}\cdot\frac{d^4(x^2-1)^4}{dx^4}=\frac{1}{8}\cdot(35x^4-30x^2+3)$$

$$P'_4=\frac{5}{2}x(7x^2-3)$$

Корни многочлена Лежандра:

$$x_{1,2}=\mp 0.33998, x_{3,4}=\mp 0.86114$$

Код решения

Производная многочлена Лежандра:

```
def p4_der(x):
    return (5/2) * x * (7 * x * x - 3)
```

Подсчет коэффицентов A_k:

```
def a_k(x):
    return 2 / ((1 - (x * x)) * np.power(p4_der(x), 2))
```

Подсчет результата интегрирования КФ Гаусса:

```
def gaussian_func(n, list_a, list_x):
    ans = 0
    for i in range(n+1):
        ans += list_a[i] * func(((a+b)/2) + (((b-a)/2) * list_x[i]))
    return ((b-a)/2) * ans
```

Производная исходной функции:

```
def eighth_derivative(x):
    return 0.8 * np.exp(x) + 0.2 * np.sin(x)
```

Вычисление погрешности (остаточного члена) интегрирования:

```
def error(n):
    err1 = (np.power(2, 2*n+3) / ((2*n+3) * np.math.factorial(2*n+2))) *
(np.power(np.power(np.math.factorial(n+1), 2) / np.math.factorial(2*n+2), 2)) * max_der_6()
    return err1 * np.power((b-a)/2, 2*n+3)
```

Результат работы программы

```
Roots of Legendre: [-0.339981043584856, 0.339981043584856, -0.861136311594053, 0.861136311594053]
A_k: [0.6521451548625462, 0.6521451548625462, 0.3478548451374528, 0.3478548451374528]
Integration result: 3.244066987775631
R_n <= 2.8313669204785735e-09
Real integral res: 3.244066989544434
Real error: 1.768802881940701e-09
```

Выводы

Для вычисления интеграла с помощью КФ Гаусса сначала были вычислены корни многочлена Лежандра, затем вычислены коэффиценты для КФ Гаусса, и наконец, вычислен интеграл заданным методом с точностью (оцененной сверху) $R_n \leq 2.8313669 \cdot 10^{-9}$.

Часть 3. Сравнение результатов вычисления в частях 1 и 2

Сравнить полученные результаты.

В первой части было необходимо найти такой шаг, чтобы точность интеграла была 10^{-5}. Мы нашли шаг и оценили погрешность по формуле трапеций, и получили погрешность $|R| \le 9.9735 \cdot 10^{-6}$. Погрешность по формуле мы оценивали сверху, брав максимальное значение производной, из-за чего оценка истинной погрешности и погрешности по формуле различаются.

Во второй части был вычислен интеграл с помощью КФ Гаусса, истинная погрешность составила $1.768803 \cdot 10^{-9}$, а оценка погрешности по формуле с подстановкой максимального значения производной составила $R_n \le 2.8313669 \cdot 10^{-9}$.

Как можно увидеть, КФ Гаусса дает гораздо более точный результат при меньшем количестве итераций (у Гаусса n=3, у метода трапеций n=197). При подсчете КФ Гаусса мы сначала высчитываем коэффиценты и значения многочлена Лежандра в узлах, которые помогают свести количество итераций метода к минимуму. При увеличении АСТ точность численного интегрирования возрастает, так как мы получаем большее количество отрезков, следовательно погрешность (остаточный член интегрирования) уменьшается.