

ЧИСЛЕННЫЕ МЕТОДЫ МАТЕМАТИЧЕСКОЙ ФИЗИКИ

ЛАБОРАТОРНАЯ РАБОТА 2

Методы решения граничной задачи для ОДУ-2

Крачковский Даниил
5 группа

Преподаватель:
Будник Анатолий Михайлович

7 июня 2018 г.

Постановка задачи

Дана задача ОДУ-2:

$$((2-x)u'(x))' - xu(x) = \sin(x) - 2\cos(x), \quad x \in [0, 1]$$

$$2u'(0) = u(0) - 1$$

$$-u'(1) = \operatorname{tg}(1)u(1)$$

Решить данную граничную задачу ОДУ-2 методом баланса и методом Рунца.

Алгоритм

Метод баланса

Запишем схему, полученную данным методом, аппроксимирующую нашу задачу:

$$\begin{aligned} \left(\frac{a_1}{h} - 1 + \frac{h}{2}d_0\right)y_0 + \left(\frac{a_1}{h}\right)y_1 &= -\left(1 + \frac{h}{2}\varphi_0\right) \\ \left(\frac{a_i}{h^2}\right)y_{i-1} - \left(\frac{a_{i+1} + a_i}{h^2} + d_i\right)y_i + \left(\frac{a_{i+1}}{h^2}\right)y_{i+1} &= -\varphi_i \quad i = \overline{1, N-1} \\ \left(\frac{a_N}{h}\right)y_{N-1} + \left(-\frac{a_N}{h} + \operatorname{tg}(1) + \frac{h}{2}d_N\right)y_N &= \frac{h}{2}\varphi_N \end{aligned}$$

где d_i , φ_i , a_i вычисляются по следующим формулам:

$$\begin{aligned} d_0 &= \frac{2}{h} \int_0^{h/2} x dx & \varphi_0 &= \frac{2}{h} \int_0^{h/2} (2\cos(x) - \sin(x)) dx \\ d_i &= \frac{1}{h} \int_{x_i-h/2}^{x_i+h/2} x dx & \varphi_i &= \frac{1}{h} \int_{x_i-h/2}^{x_i+h/2} (2\cos(x) - \sin(x)) dx \quad i = \overline{1, N-1} \\ a_i &= \left(\frac{1}{h} \int_{x_{i-1}}^{x_i} \frac{dx}{2-x}\right)^{-1} & i &= \overline{1, N} \\ d_N &= \frac{2}{h} \int_{1-h/2}^1 x dx & \varphi_N &= \frac{2}{h} \int_{1-h/2}^1 (2\cos(x) - \sin(x)) dx \end{aligned}$$

Решим образовавшуюся систему методом прогонки.

Метод Рунца

Результат

Средних:

h: 0.00894427190999916

n: 111

I: 0.6909220448382232

Симпсона:

h: 0.09306048591020996

n: 10

I: 0.47912965028132487

1 Листинг кода

```
import math

def f_x(x):
    p = 1.2
    return math.sqrt(p + x ** 2) / (1. + math.cos(p * x))

eps = 10. ** (-5)
max_f_2 = 3.
a = 0
b = 1

if __name__ == '__main__':
    h = math.sqrt(24.*eps/max_f_2)
    n = int(1. / h)
    I = h * sum(f_x(a + k*h) for k in range(0, n - 1))

    print('middle')
    print('h: {}'.format(h))
    print('n: {}'.format(n))
    print('I: {}'.format(I))
```

```

import math

def f_x(x):
    p = 1.2
    return math.sqrt(p + x ** 2) / (1. + math.cos(p * x))

eps = 10. ** (-5)
max_f_4 = 24.
a = 0
b = 1

if __name__ == '__main__':
    h = (180. * eps / max_f_4) ** (1./4.)
    n = int(1. / h)

    sum_1 = sum(f_x(a + (2*k - 1)*h) for k in range(1, int(n / 2)))
    sum_2 = sum(f_x(a + (2*k)*h) for k in range(1, int(n / 2) - 1))

    I = h / 3. * (f_x(a) + f_x(b) + 4 * sum_1 + 2 * sum_2)

    print('simpson')
    print('h: {}'.format(h))
    print('n: {}'.format(n))
    print('I: {}'.format(I))

```