МИНОБРНАУКИ РОССИИ
ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ
ВЫСШЕГО ОБРАЗОВАНИЯ
“ВОРОНЕЖСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ”

Факультет компьютерных наук

Кафедра цифровых технологий

Отчет по лабораторной №3 ЧМ

Направление 02.03.01 Математика и компьютерные науки

Студент 3-его курса 7.2 группы – Гузенко Алексей Михайлович

Вариант 3

Воронеж 2020

Содержание

Постановка задачи

Реализовать метод стрельбы для приближенного решения краевой задачи

$$y'' = y' - x, \quad y(0) = -1, \quad y(1) = -2$$

Соответствующую задачу Коши решить методом Адамса 4 порядка с шагом $h = 0.01$, а параметр вычислить методом деления отрезка пополам. Использовать точность решения $\varepsilon = 0.01$.

Приведение краевой задачи к задаче Коши

Обозначим $y' = z$

Тогда $y(0) = -1$, $z' = f(x, y, z)$, $z(0) = y'(0) = \eta$

Введем функцию ошибки $\Phi(\eta) = y(1, \eta) - (-2)$, где $y(x, \eta)$ – решение краевой задачи при параметре η

Схема вычисления значений x_i, y_i, z_i для используемого метода решения задачи Коши

$$x_{i+1} = x_i + h, \ \ y_1 = y_0 + h\mathrm{f}(x_0, y_0, z_0), \quad z_1 = z_1 + hf(x_0, y_0, z_0)$$

$$y_2 = y_1 + h/2 * \left(3 * f(x_1, y_1, z_1) - f(x_0, y_0, z_0)\right)$$

$$z_2 = z_1 + h/2 * \left(3 * f(x_1, y_1, z_1) - f(x_0, y_0, z_0)\right)$$

$$y_3 = y_2 + h/12 * \left(23 * f(x_2, y_2, z_2) - 16 * f(x_1, y_1, z_1) + 5 * f(x_0, y_0, z_0)\right)$$

$$\mathrm{z}_3 = \mathrm{z}_2 + h/12 * \left(23 * f(x_2, y_2, z_2) - 16 * f(x_1, y_1, z_1) + 5 * f(x_0, y_0, z_0)\right)$$

$$y_4 = y_3 + h/24 * \left(55 * f(x_3, y_3, z_3) - 59 * f(x_2, y_2, z_2) + 36 * f(x_1, y_1, z_1) - 9 * f(x_0, y_0, z_0)\right)$$

$$z_4 = z_3 + h/24 * \left(55 * f(x_3, y_3, z_3) - 59 * f(x_2, y_2, z_2) + 36 * f(x_1, y_1, z_1) - 9 * f(x_0, y_0, z_0)\right)$$

$$y_i = y_{i-1} + h/24 * \left(55 * f(x_{i-1}, y_{i-1}, z_{i-1}) - 59 * f\left(x_{i-2}, y_{i_2}, z_{i-2}\right) + 36 * f(x_{i-3}, y_{i-3}, z_{i-3}) - 9 * f(x_{i-4}, y_{i-4}, z_{i-4})\right)$$

$$\mathrm{z}_i = \mathrm{z}_{i-1} + h/24 * \left(55 * f(x_{i-1}, y_{i-1}, z_{i-1}) - 59 * f\left(x_{i-2}, y_{i_2}, z_{i-2}\right) + 36 * f(x_{i-3}, y_{i-3}, z_{i-3}) - 9 * f(x_{i-4}, y_{i-4}, z_{i-4})\right)$$

$$x_0 = a, \qquad y_0 = b, \qquad z_o = \eta$$

Выбор первоначальных значений параметра стрельбы

$$\eta_1 = 0$$

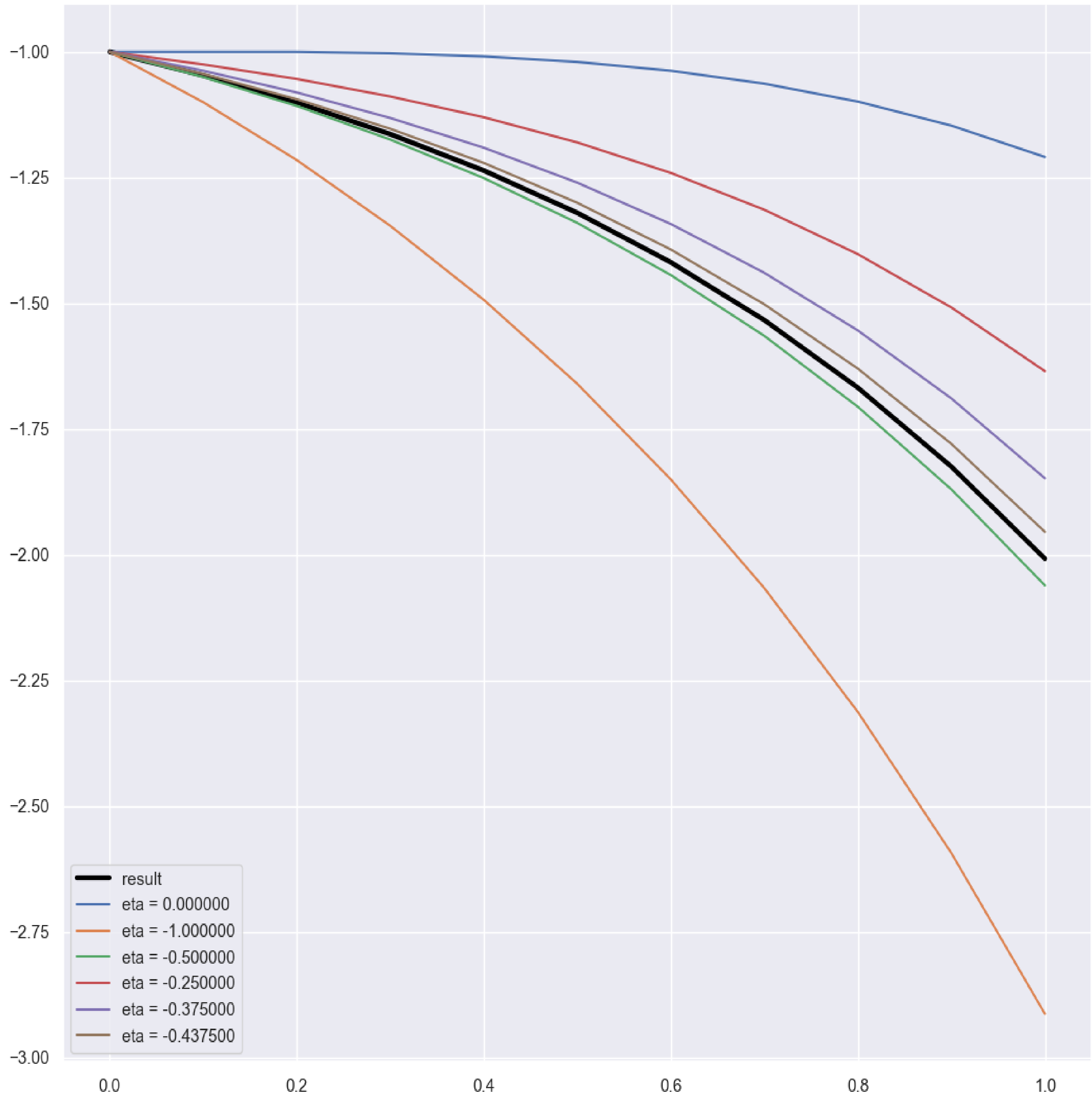
$$\eta_2 = \frac{B - A}{b - a} = \frac{-2 + 1}{1 - 0} = -1$$

Исходный код решения (Python 3.8)

```
import matplotlib.pyplot as plt
import numpy as np
import seaborn

a, b = 0, 1
A, B = -1, -2


def f(x, u):
    return u[1] - x

def F(f):
  return lambda x, u: np.append(u[1:], f(x, u))

def adams4(f, a, b, u0, h):
    func = F(f)
    x = a
    u = np.array(u0)
    res = [(x, u[0])]
    prev = [(x, u)]
    while x + h <= b:
        if len(prev) == 1:
            u = u + h * func(*prev[-1])
        elif len(prev) == 2:
            u = u + h / 2 * (3 * func(*prev[-1]) - func(*prev[-2]))
        elif len(prev) == 3:
            u = u + h / 12 * (23 * func(*prev[-1]) - 16 * func(*prev[-2]) + 5 * func(*prev[-
3]))
        else:
            u = u + h / 24 * (55 * func(*prev[-1]) - 59 * func(*prev[-2]) + 37 * func(*prev[-
3]) - 9 * func(*prev[-4]))
        x = x + h
        res.append((x, u[0]))
        prev.append((x, u))
    return res


def err(actual, res):
  return actual - res[-1][1]

eta_1 = 0
eta_2 = (B - A) / (b - a)

def solve(f, a, b, A, B, eta_1, eta_2, h=0.01, eps=0.01):
  eta = (eta_1 + eta_2) / 2
  hist = [
      (eta_1, adams4(f, a, b, [A, eta_1], h)),
      (eta_2, adams4(f, a, b, [A, eta_2], h))
  ]
  errors = {
      eta_1: err(B, hist[0][1]),
      eta_2: err(B, hist[1][1])
  }
  hist.append((eta, adams4(f, a, b, [A, eta], h)))
  errors[eta] = err(B, hist[-1][1])
  while (abs(errors[eta]) > eps):
    if (errors[eta] * errors[eta_1] < 0):
      eta_1, eta_2 = min(eta, eta_1), max(eta, eta_1)
    else:
      eta_1, eta_2 = min(eta, eta_2), max(eta, eta_2)
    eta = (eta_1 + eta_2) / 2
    hist.append((eta, adams4(f, a, b, [A, eta], h)))
```

```python
    errors[eta] = err(B, hist[-1][1])
  return hist[-1], hist

res, hist = solve(f, a, b, A, B, eta_1, eta_2, h=0.1, eps=0.01)
arr = np.array(res[1])
seaborn.set()
plt.rcParams["figure.figsize"] = (20, 15)

fig, ax = plt.subplots()

arr = np.array(res[1])
x = arr[:, 0]
y = arr[:, 1]
line = ax.plot(x, y, label='result', linewidth=3, color='black')

for eta, r in hist[:-1]:
  arr = np.array(r)
  x = arr[:, 0]
  y = arr[:, 1]
  line = ax.plot(x, y, label='eta = {:f}'.format(eta))

ax.legend()
plt.show()
```

График, отражающий результат каждой “стрельбы”



Полученное приближенное решение в форме таблицы

η	$y(b, \eta)$
0	-1.20893811
-1.0	-2.91287621
-0.5	-2.06090716
-0.25	-1.63492263
-0.375	-1.8479149
-0.4375	-1.95441103
-0.46875	-2.0076590946865616