Министерство науки и высшего образования Российской Федерации
Федеральное государственное бюджетное образовательное учреждение высшего образования
«Московский государственный технический университет имени Н.Э. Баумана
(национальный исследовательский университет)»
(МГТУ им. Н.Э. Баумана)

ФАКУЛЬТЕТ Информатики и систем управления

КАФЕДРА Теоретической информатики и компьютерных технологий

Отчет по лабораторной работе № 4
«Сравнительный анализ методов численного решения краевой задачи для линейного дифференциального уравнения второго порядка»

по курсу «Численные методы»

Выполнил:

студент группы ИУ9-61Б

Баев Д. А.

Проверила:

Домрачева А. Б.

Москва, 2023

1. **Цель**

Целью данной работы является сравнение по точности решения двух методов численного решения краевой задачи для линейного дифференциального уравнения второго порядка:

1. Метод прогонки
2. Метод стрельбы

2. **Постановка задачи**

Дано: краевая задача для линейного дифференциального уравнения (ДУ) второго порядка

$y'' + p(x)y' + q(x)y = f(x)$

$y(0) = a$

$y(1) = b$

Найти: аналитическое решение задачи Коши:

$y'' + p(x)y' + q(x)y = f(x),\ y(0) = y_0,\ y'(0) = y_0'$;

По найденному решению задачи Коши вычислить $b = y(1)$;

С помощью метода прогонки и метода стрельбы найти численное решение $\left(x_i, y_i\right),\ i = \overline{0, n},\ n = 10$ краевой задачи для того же уравнения с краевыми условиями $y(0) = a,\ y(1) = b$;

Для каждого метода вычислить $\left|y_i - \widetilde{y}_i\right|,\ i = \overline{0, n},$ найти погрешность численного решения $\|y - \widetilde{y}\| = max\left|y_i - \widetilde{y}_i\right|$ и сравнить результаты этих методов.

Индивидуальный вариант:

$p = -7,\ q = 12,\ f(x) = 5,\ y_0 = 1,\ y_0' = 2$

3. **Основные теоретические сведения**

Метод прогонки

Пусть требуется решить краевую задачу на отрезке $[0, 1]$. Тогда отрезок разбивается на n равных отрезков длины $h = \frac{1}{n}$.

Приближенный численным решением краевой задачи для ДУ второго порядка называется сеточная функция $\left(x_i, y_i\right)$, $i = \overline{0, n}$, заданная в точках $x_i = ih$, $h = \frac{1}{n}$.

Обозначим значения коэффициентов уравнения в точках x_i, $i = \overline{0, n}$ через $p_i = p\left(x_i\right)$, $q_i = q\left(x_i\right)$, $f_i = f(x_i)$.

Применяя разностную аппроксимацию производных по формулам численного дифференцирования получаем приближенную систему уравнений относительно ординат сеточной функции:

$$\frac{y_{i+1} - 2y_i + 2y_{i-1}}{h^2} + p_i \frac{y_{i+1} - y_{i-1}}{2h} + q_i y_i = f_i$$

После преобразования система имеет вид:

$$y_{i-1}\left(1 - \frac{h}{2}p_i\right) + y_i\left(h^2 q_i - 2\right) + y_{i+1}\left(1 + \frac{h}{2}p_i\right) = h^2 f_i, \ i = \overline{1, n-1}$$

с краевыми условиями $y_0 = a$, $y_n = b$

Данная система имеет порядок $n - 1$ и представляет собой трехдиагональную систему линейных алгебраических уравнений. Ее следует решать методом прогонки.

Метод стрельбы

Пусть требуется решить краевую задачу на отрезке $[0, 1]$. Тогда отрезок разбивается на n равных отрезков длины $h = \frac{1}{n}$.

Приближенный численным решением краевой задачи для ДУ второго порядка называется сеточная функция $\left(x_i, y_i\right)$, $i = \overline{0, n}$, заданная в точках $x_i = ih$, $h = \frac{1}{n}$.

Обозначим значения коэффициентов уравнения в точках x_i, $i = \overline{0, n}$ через $p_i = p\left(x_i\right)$, $q_i = q\left(x_i\right)$, $f_i = f(x_i)$.

Заменим производные во внутренних точках их разностными аналогами:

$$y_i' = \frac{y_{i+1} - y_{i-1}}{2h},\ \ y_i'' = \frac{y_{i+1} - 2y_i + y_{i-1}}{h^2},\ \ i = \overline{1, n-1}$$

Обозначим значения $y_0(x_i) = y_0[i],\ \ y_1(x_i) = y_1[i]$.

Будем искать решения, удовлетворяющие условиям:

$$y_0[0] = a,\ \ y_0[1] = D_0,\ \ y_1[0] = 0,\ \ y_1[1] = D_1 \neq 0$$

($y_0(x_i) = y_0[i]$ – частное решение, $y_1(x_i) = y_1[i]$ – часть общего решения)

где $D_0 = a + O(h)$, $D_1 = O(h)$. Пусть $O(h) = h$.

Для определения y_0 и y_1 получаем уравнения:

$$\frac{y_0[i+1] - 2y_0[i] + y_0[i-1]}{h^2} + p_i \frac{y_0[i+1] - y_0[i-1]}{2h} + q_i y_0[i] = f_i,$$

$$\frac{y_1[i+1] - 2y_1[i] + y_1[i-1]}{h^2} + p_i \frac{y_1[i+1] - y_1[i-1]}{2h} + q_i y_1[i] = 0.$$

Используя начальные условия, имеем:

$$y_0[i+1] = \frac{\left(f_i h^2 + \left(2 - q_i h^2\right) y_0[i] - \left(1 - p_i \frac{h}{2}\right) y_0[i-1]\right)}{1 + \frac{p_i h}{2}},$$

$$y_1[i+1] = \frac{\left(\left(2 - q_i h^2\right) y_1[i] - \left(1 - p_i \frac{h}{2}\right) y_1[i-1]\right)}{1 + \frac{p_i h}{2}},$$

$$i = \overline{1, n-1};$$

Далее последовательно определяем $y_0[2], \dots, y_0[n], y_1[2], \dots, y_1[n]$ и находим $C_1 = \left(b - y_0[n]\right) / y_1[n]$.

Искомое решение ищется по формулам:

$$y[i] = y_0[i] + C_1 y_1[i],\ \ \ i = \overline{1, n}.$$

4. Реализация

Аналитическое решение для задачи Коши:

$y(x) = \frac{1}{12}(4e^{3x} + 3e^{4x} + 5)$

Теперь можно найти b:

b = 20.761383149348614

Листинг 1. Реализация метода прогонки и метода стрельбы для решения краевой задачи для ДУ второго порядка.

```
import math
import pandas as pd


def f(x):
    return 5


def check(x):
    return 1/12 * (4 * math.exp(3*x) + 3 * math.exp(4*x) + 5)


def gun(n, a, b, h, p, q, x_0, o_h):
    y_0 = [0] * (n + 1)
    y_1 = [0] * (n + 1)
    y_0[0] = a
    y_0[1] = a + o_h
    y_1[1] = o_h

    for i in range(1, n):
        y_0[i + 1] = (h * h * f(x_0 + i * h) - (1 - h / 2 * p) * y_0[i - 1] -
(h * h * q - 2) * y_0[i]) / (1 + h / 2 * p)
        y_1[i + 1] = ((h / 2 * p - 1) * y_1[i - 1] - (h * h * q - 2) *
y_1[i]) / (1 + h / 2 * p)

    if y_1[n] == 0:
        return gun(n, a, b, h, p, q, x_0, o_h + 1)
    else:
        c1 = (b - y_0[n]) / y_1[n]
        return [y_0[i] + c1 * y_1[i] for i in range(n+1)]


def run(mid, top, bot, b):
    n = len(b)
    x = [0] * n
    v = [0] * n
    u = [0] * n

    v[0] = -top[0] / mid[0]
    u[0] = b[0] / mid[0]
    for i in range(1, n):
        v[i] = -top[i] / (bot[i] * v[i - 1] + mid[i])
        u[i] = (b[i] - bot[i] * u[i - 1]) / (bot[i] * v[i - 1] + mid[i])
```

```
    x[n - 1] = u[n - 1]
    for i in range(n - 1, 0, -1):
        x[i - 1] = v[i - 1] * x[i] + u[i - 1]
    return x


def make_equation(h, p, q, n, x_0, a, b):
    bot = []
    mid = []
    top = []
    result = []

    mid.append(h * h * q - 2)
    top.append(1 + h / 2 * p)
    result.append(h * h * f(x_0 + h) - a * (1 - h / 2 * p))
    bot.append(0)

    for i in range(2, n - 1):
        bot.append(1 - h / 2 * p)
        mid.append(h * h * q - 2)
        top.append(1 + h / 2 * p)
        result.append(h * h * f(x_0 + h * i))

    bot.append(1 - h / 2 * p)
    mid.append(h * h * q - 2)
    result.append(h * h * f(x_0 + (n - 1) * h) - b * (1 + h / 2 * p))
    top.append(0)

    return bot, mid, top, result


if __name__ == "__main__":
    p = -7
    q = 12
    n = 10
    x_0 = 0
    x_n = 1
    h = (x_n - x_0) / n
    a = check(x_0)
    b = check(x_n)
    print(b)

    bot, mid, top, result = make_equation(h, p, q, n, x_0, a, b)
    x_arr = [x_0 + i * h for i in range(n + 1)]
    y_true_arr = [check(x_0 + i * h) for i in range(n + 1)]
    y_ev_arr = [a] + run(mid, top, bot, result) + [b]
```

```
    delta = [y_true_arr[i] - y_ev_arr[i] for i in range(n + 1)]
    y_gun = gun(n, a, b, h, p, q, x_0, h)
    delta_gun = [y_true_arr[i] - y_gun[i] for i in range(n + 1)]

    #pd.options.display.float_format = '{:,.13f}'.format
    #pd.set_option('display.max_columns', None)
    answer = pd.DataFrame({"x": x_arr, "y_true": y_true_arr, "y_ev": y_ev_arr,
"diff": delta, "gun": y_gun, "delta_gun": delta_gun})
    print(answer)
    print()
    for i in range(n + 1):
        print(delta_gun[i] - delta[i])
```

5. Тестирование

Таблица 1 - Результаты метода прогонки

Значение x	Значение y	Значение $\tilde{y}$	$\lvert y_i - \tilde{y}_i \rvert$
0	1.000000	1.000000	0
0.1	1.239576	1.225316	0.014259
0.2	1.580425	1.543992	0.036433
0.3	2.066564	1.997736	0.068828
0.4	2.761630	2.648236	0.113394
0.5	3.757827	3.587294	0.170533
0.6	5.189010	4.952297	0.236712
0.7	7.249885	6.949958	0.299927
0.8	10.224191	9.892799	0.331392
0.9	14.526136	14.255414	0.270722
1.0	20.761383	20.761383	0

$$\|y - \tilde{y}\| = max\lvert y_i - \tilde{y}_i \rvert = 0.331392$$

Таблица 2 - Результаты метода стрельбы

Значение x	Значение y	Значение $\tilde{y}$	$\lvert y_i - \tilde{y}_i \rvert$
0	1.000000	1.000000	0.000000e+00
0.1	1.239576	1.225316	1.425931e-02
0.2	1.580425	1.543992	3.643258e-02
0.3	2.066564	1.997736	6.882792e-02
0.4	2.761630	2.648236	1.133942e-01

0.5	3.757827	3.587294	1.705333e-01
0.6	5.189010	4.952297	2.367125e-01
0.7	7.249885	6.949958	2.999271e-01
0.8	10.224191	9.892799	3.313924e-01
0.9	14.526136	14.255414	2.707222e-01
1.0	20.761383	20.761383	3.552714e-15

$$\|y - \tilde{y}\| = max\left|y_i - \tilde{y}_i\right| = 3.313924e - 01$$

Хотя из данных таблицы можно сделать вывод, что погрешность двух методов является одинаковой, это не так. Ниже приведена разница погрешностей между двумя методами:дами

Таблица 3 - Разница погрешностей (метод стрельбы - метод прогонки)

Значение x	Разница погрешности
0	0.0
0.1	-2.220446049250313e-16
0.2	-2.220446049250313e-16
0.3	-2.220446049250313e-16
0.4	-4.440892098500626e-16
0.5	-8.881784197001252e-16
0.6	-1.7763568394002505e-15
0.7	-2.6645352591003757e-15
0.8	-5.329070518200751e-15
0.9	-1.7763568394002505e-15
1.0	3.552713678800501e-15

Можно заметить достаточно большие вычислительные погрешности обоих методов. Это возникает из-за слишком малого числа разбиений (всего 10). Проверим результат работы метода прогонки при числе разбиений 50.

Таблица 4 - Результаты метода прогонки при n = 50

Значение x	Значение y	Значение $\tilde{y}$	$\left\|y_i - \tilde{y}_i\right\|$
0	1.000000	1.000000	0
0.1	1.239576	1.239001	0.000575
0.2	1.580425	1.578967	0.001458
0.3	2.066564	2.063832	0.002732
0.4	2.761630	2.757170	0.113394

0.5	3.757827	3.751183	0.006644
0.6	5.189010	5.179881	0.009129
0.7	7.249885	7.238444	0.011441
0.8	10.224191	10.211696	0.012495
0.9	14.526136	14.516053	0.010083
1.0	20.761383	20.761383	0

Для метода стрельбы ситуация идентична.

6. Вывод

В процессе выполнения лабораторной работы были использованы два метода приближенного численного решения краевой задачи для дифференциального уравнения второго порядка: метод прогонки и метод стрельбы.

Сравнение результатов работы показало, что метод стрельбы обладает более высокой точностью, но разница между ним и методом прогонки невелика и может быть не очевидна. Даже при использовании случайных значений в методе стрельбы ($D_0 = a + O(h)$, $D_1 = O(h)$) было достигнуто более точное решение, чем в методе прогонки без таких значений.

Также было обнаружено, что при малом числе разбиений возникает серьезная вычислительная погрешность, но при увеличении числа разбиений погрешность существенно уменьшается.