Санкт-Петербургский политехнический университет
Высшая школа прикладной математики
и вычислительной физики, ФизМех

Направление подготовки
«Прикладная математика и информатика»

Отчет по лабораторной работе №7
«Численнные методы решения краевых задач для обыкновенных дифференциальных уравнений»

Выполнил студент гр. 5030102/00003 Петрошенко А.В.

Преподаватель: Курц В.В.

Санкт-Петербург
2021

Формулировка задачи и ее формализация

Дано ОДУ 2-го порядка с переменными коэффициентами:

$$p(x)y'' + q(x)y' + r(x) = f(x) \tag{1}$$

$p(x), q(x), r(x), f(x) \in C([a,b])$ Дифференциальный оператор $L = p\dfrac{d^2}{dx^2} + q\dfrac{d}{dx} + r$. Тогда (1) $\Leftrightarrow L(y) = f$
Общий вид граничных условий:

$$\begin{cases} \alpha_0 y(a) + \alpha_1 y'(a) = A \\ \beta_0 y(b) + \beta_1 y'(b) = B \end{cases} \tag{2}$$

$\alpha_0^2 + \alpha_1^2 \neq 0$, $\beta_0^2 + \beta_1^2 \neq 0$
И дополнительное условие $\beta_1 \neq 0$ постановка задачи:
Необходимо решить краевую задачу на отрезке $[0,1]$ вида:

$$\begin{cases} (e^x + 1)y'' - y' - e^x y = e^x \\ -y'(a) = -1 \\ y(b) + y'(b) = 2e - 1 \end{cases} \tag{3}$$

т.е. $\alpha_0 = 1$, $\alpha_1 = -1$, $\beta_0 = 1$, $\beta_1 = 1$
Нужно вычислить точное решение $y^*(x) = e^x - 1$

Алгоритм метода и условия его применимости

Заменим производные конечными разностями:

$$\frac{dy}{dx} = \frac{y_{i+1} - y_{i-1}}{2h} \tag{4}$$

Тогда вторая производная будет иметь вид:

$$\frac{d^2y}{dx^2} = \frac{y_{i+1} - 2y_i + y_{i-1}}{h^2} \tag{5}$$

Теперь вместо ДУ будем решать СЛАУ:

$$\begin{cases} \alpha_0 y_0 + \alpha_1 \dfrac{-3y_0 + 4y_1 - y_2}{2h} = A \\ p_k \dfrac{y_{k+1} - 2y_k + y_{k-1}}{h^2} + q_k \dfrac{y_{k+1} - y_{k-1}}{2h} + r_k y_k = f_k \\ \alpha_0 y_0 + \alpha_1 \dfrac{3y_n - 4y_{n-1} + y_{n-2}}{2h} = B \end{cases} \tag{6}$$

Данная матрица практически трехдиагональная, за исключением первой и последней строчки. Приведем ее к трехдигональной вычитанием соседних строк и решим ее, например, алгоритмом прогонки. Получим вектор Y, который и будет решением исходного ОДУ. Условия применимости

1. $\alpha_0^2 + \alpha_1^2 \neq 0$, $\beta_0^2 + \beta_1^2 \neq 0$, $\beta_1 \neq 0$
2. По теореме должны выполнятся 3 условия:

$$\begin{cases} p(x) \geq 0 \\ p(x) \geq \frac{h}{2}|q(x)| \\ r(x) \leq 0 \end{cases}$$

3. Так как мы будем решать трехдиагональную матрицу, то она должна обладать диагональным преобладанием для устойчивости метода прогонки

Предварительный анализ задачи

1. Из выбора коэффициентов данные условия выполнены по построению
2. Все 3 условия теоремы выполнены:

$$\begin{cases} e^x + 1 \geq 0 \\ e^x + 1 \geq \frac{h}{2}|-1| \\ -e^x \leq 0 \end{cases}$$

3. Промежуточные строки матрицы(не первая и не последняя) обладают диагональным преобладанием: $e^{x_i}h^2 + 2(e^{x_i}+1) \geq (-(e^{x_i}+1) - \dfrac{h}{2}) + (\dfrac{h}{2} - (e^{x_i}+1)) = -2(e^{x_i}+1)$. Т.к. при приведении к трехдиагональной матрицы мы из не дигонального элемента вычитали диагональный и наоборот, т.е. из большего меньшее и из меньшего большее, то нам достаточно проверить исходные коэффициенты, чтобы первая и последняя строки тоже обладали диагональным преобладанием: $\alpha_0 h - \dfrac{3}{2}\alpha_1 = h + \dfrac{3}{2} \geq -2 = 2\alpha_1$

Тестовый пример для задач малой размерности

Протестируем метод, разбив данный отрезок $[0, 1]$ на 5 частей. Тогда $k = 5$, $h = \dfrac{b-a}{k} = 0.2$
Точное решение: $y^*(x) = e^x - 1$

x	0	0.2	0.4	0.6	0.8	1
y	0	0.221	0.492	0.822	1.225	1.718

Построим матрицу по формулам (6) и сразу преобразуем ее в трехдигональную:

$$A = \begin{pmatrix} 1.153 & -0.941 & 0 & 0 & 0 & 0 \\ -2.321 & 4.492 & -2.121 & 0 & 0 & 0 \\ 0 & -2.592 & 5.043 & -2.392 & 0 & 0 \\ 0 & 0 & -2.922 & 5.717 & -2.722 & 0 \\ 0 & 0 & 0 & -3.325 & 6.540 & -3.125 \\ 0 & 0 & 0 & 0 & -1.017 & 1.230 \end{pmatrix} \tag{7}$$

Вектор свободных членов F будет выглядеть так:

$$F = (-0.211, -0.049, -0.060, -0.073, -0.090, 0.874) \tag{8}$$

Теперь решаем эту СЛАУ методом прогонки и получаем вектор Y:

$$Y = (-0.003, 0.221, 0.494, 0.827, 1.234, 1.730) \tag{9}$$

Как видно значения практически совпадают, график точного и численного решения для 5 частей представлен ниже.

Контрольные тесты

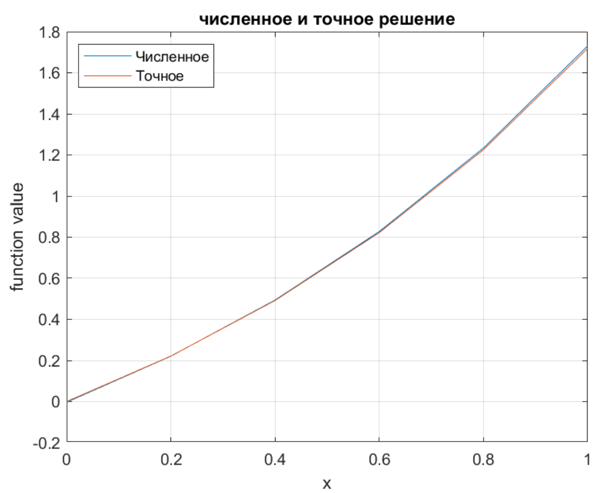
1. Разобъем отрезок $[0, 1]$ на 5 частей, решим ДУ и посмотрим на графики точного и численного решения, а так же на график ошибки.
2. Будем разбивать наш отрезок на части (от 10 до 500 с шагом 1) каждый раз решая ДУ и вычисляя норму вектора погрешности.
3. Разобъем отрезок $[0, 1]$ на 10 частей и будем вносить в граничные условия возмущения различного порядка(от 10^{-10} до 1) относительно порядка значения условия

Модульная структура программы

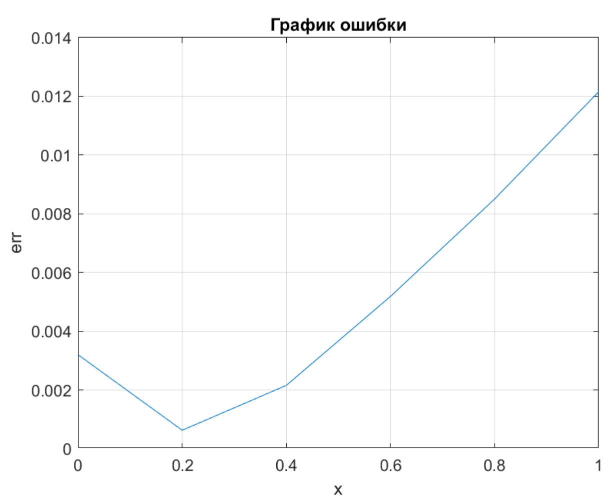
```
double p(double x)
double q(double x)
double r(double x)
double f(double x)
```
-Функции переменных коэффициентов.
```
vector<vector<double>> CreateMatrix(double a, double b, int n, double alpha0,
double alpha1, double beta0, double beta1, double A, double B);
```
-Функция задания матрицы.
```
vector<double> ThomasAlgorithm(vector<double> w0, vector<double> w1,
vector<double> w2, vector<double> F);
```
-Алгоритм прогонки, возвращающий вектор Y.

Численный анализ

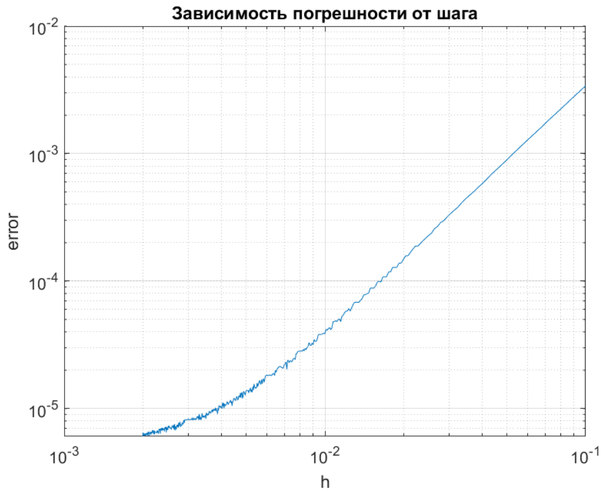
▷ Точное и численное решение:



Оба графика накладываются друг на друга, что показывает, что решения практически совпадают.
▷ Ошибка численного решения:

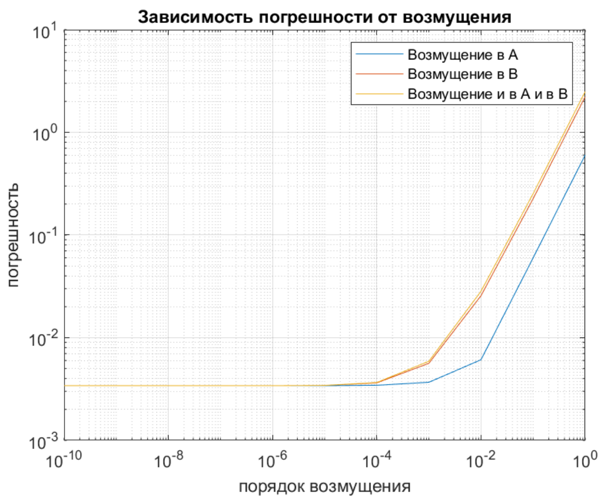


В граничных точках ошибка ненулевая, т.к. в условии присутствовала производная. Значит минимум ошибки будет где-то между границами отрезка, так оно и есть.
▷ Погрешности:



Порядок погрешности совпадает с порядком метода. На малых h происходят колебания, которые вызвана вычислительной погрешностью алгоритма прогонки, поскольку матрица становится большой.

▷ Возмущение:



Мы рассмотрели 3 возмущения: 2 только в одном граничном условии и 1 в обоих. При малых возмущениях погрешность не меняется, но начиная с 10^{-5} они отклоняются, . Так же можно рассмотреть противоположные возмущения для каждого случая, но на картину это повлияет не сильно

Общие выводы

В данной лабораторной работе мы научились численно решать ОДУ 2-го порядка на заданном промежутке с помощью метода конечных разностей 2-го порядка с ненулевым коэффициентом β_1. Реализация метода довольно трудная, т.к. нужно построить матрицу, привести ее к трехдиагональной и решить. К тому же метод требует довольно много вычислений, что влияет на конечную погрешность.