Санкт-Петербургский политехнический университет

Высшая школа прикладной математики и вычислительной физики, ИПММ

Направление подготовки

«01.03.02 Прикладная математика и информатика»

Отчет по лабораторной работе №**5**

**тема "Решение задачи Коши методами Рунге-Кутты"**

**дисциплина "Численные методы"**

Выполнил студент гр. **3630102/80001** Д.В. Хрипунков

Преподаватель: С.Б. Добрецова

Санкт-Петербург

**2020**

1. Формулировка задачи

Необходимо решить задачу Коши с использованием формулы Эйлера-Коши с заданной точностью

По результатам решения найти и проанализировать зависимости:

1. Фактической точности от задаваемой точности , где заданное точное значение , а вычисленное с помощью метода Эйлера-Коши значение
2. Число итераций от заданной точности . Под числом итераций подразумевается число разбиений шага пополам. Для разных точек при одной точности оно может быть разным, поэтому считается наибольшее
3. Фактической погрешности от величины ошибки . Ошибка вносится в условие

По зависимостям построить графики в MATLAB и установить закономерности

2. Постановка задачи

Дано ; По формуле необходимо с заданной точностью построить сеточную функцию , аппроксимирующую решение данной задачи Коши.

3. Алгоритм метода

Дано , , для решения задачи Коши с заданной точностью используем следующий алгоритм:

1. Построить равномерную сетку для всего отрезка : , задать нужное , установить
2. Вычислить по формуле
3. Разделить на и провести те же вычисления, переходя от последовательно к промежуточным точкам между , а потом к самой точке
4. Провести проверку . Если это верно, то значение в точке вычислено с нужной точностью и можно переходить к следующей точке с пункта 2. Если проверка не пройдена, то повторять пункты 3-4 до тех пор, пока не будет пройдена проверка

4. Анализ задачи

В качестве тестовой функции используем . Для применения метода достаточно чтобы решение задачи Коши существовало на всём промежутке. Для этого достаточно чтобы функция была непрерывна.

5. Тестовый пример

В качестве тестового примера возьмём , ; точки , точное решение , начальный шаг

1. i = 0:

При

- точность

1. i = 1:

При

– точность

Теперь сравним полученные результаты с исходными данными:

Фактическая точность решения оказалась выше, чем предполагаемая.

6. Контрольные тесты

Весь анализирующий функционал реализован на языке c++ в Visual Studio 2019, а графическая интерпретация в пакете MATLAB.

Во всех тестах сетка равномерная, .

Во всех тестах на зависимость фактической точности и числа итераций от задаваемой точности , значение будет изменяться от до с шагом в .

Во всех тестах на зависимость фактической точности от внесённой ошибки значение задаваемой точности будет фиксированным и , а значение будет меняться от до с шагом в 0.1.

7. Модульная структура программы

double CountPreciseSolution(double x) – функция вычисления точного значения заданной функции , принимающая на вход значение

double CountDevar(double x, double y) – функция вычисления точного значения заданной функции , принимающая на вход значения

void FillGrid(vector <double>& xi) – функция заполнения массива равномерной сетки , принимающая границы отрезка и шаг в качестве глобальных аргументов

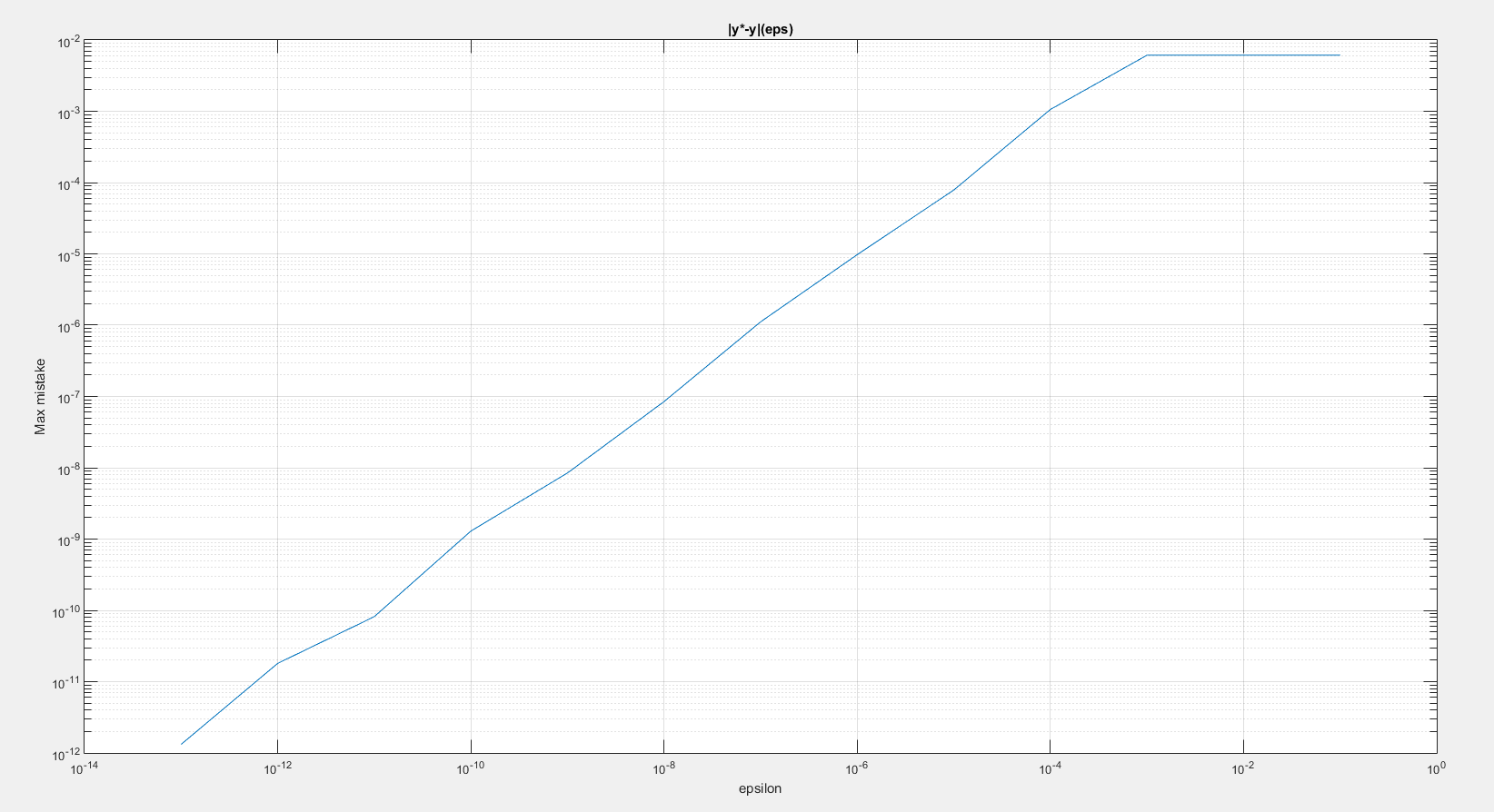
double CountNextY(double x, double y, double h) – функция вычисления значения по формуле метода, принимающая текущие

unsigned FindSolution(vector <double> xi, vector <double>& yi) – функция заполнения массива решений с заданной точностью, который принимает границы отрезка, значение , числа точек в сетке n, шага h и точности в качестве глобальных аргументов

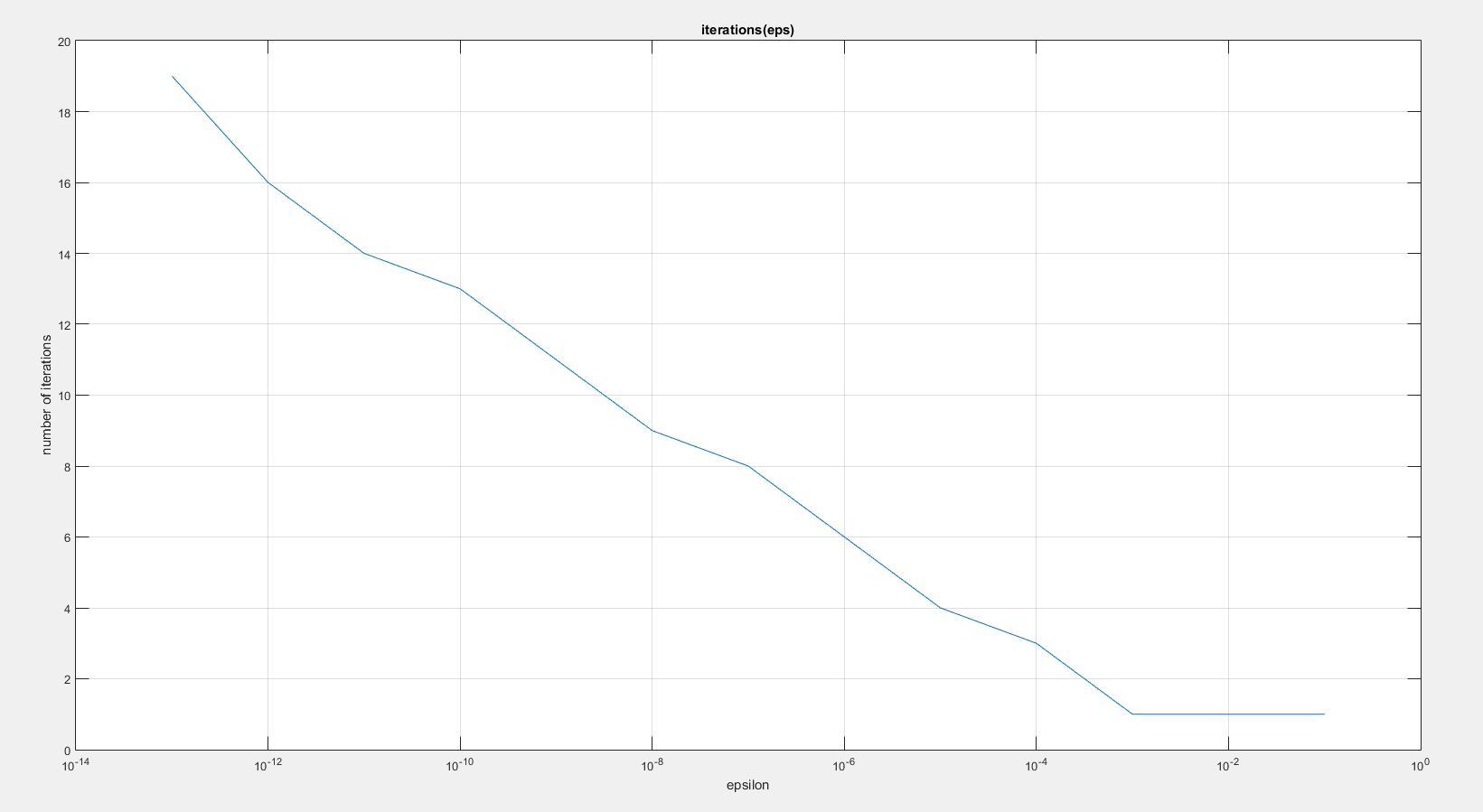
double FindMaxMistake(vector <double> y, vector <double> yi) – функция вычисления фактической точности как , которая принимает массив значений и как входные параметры

8. Численный анализ

Графики зависимостей, построенные в MATLAB:

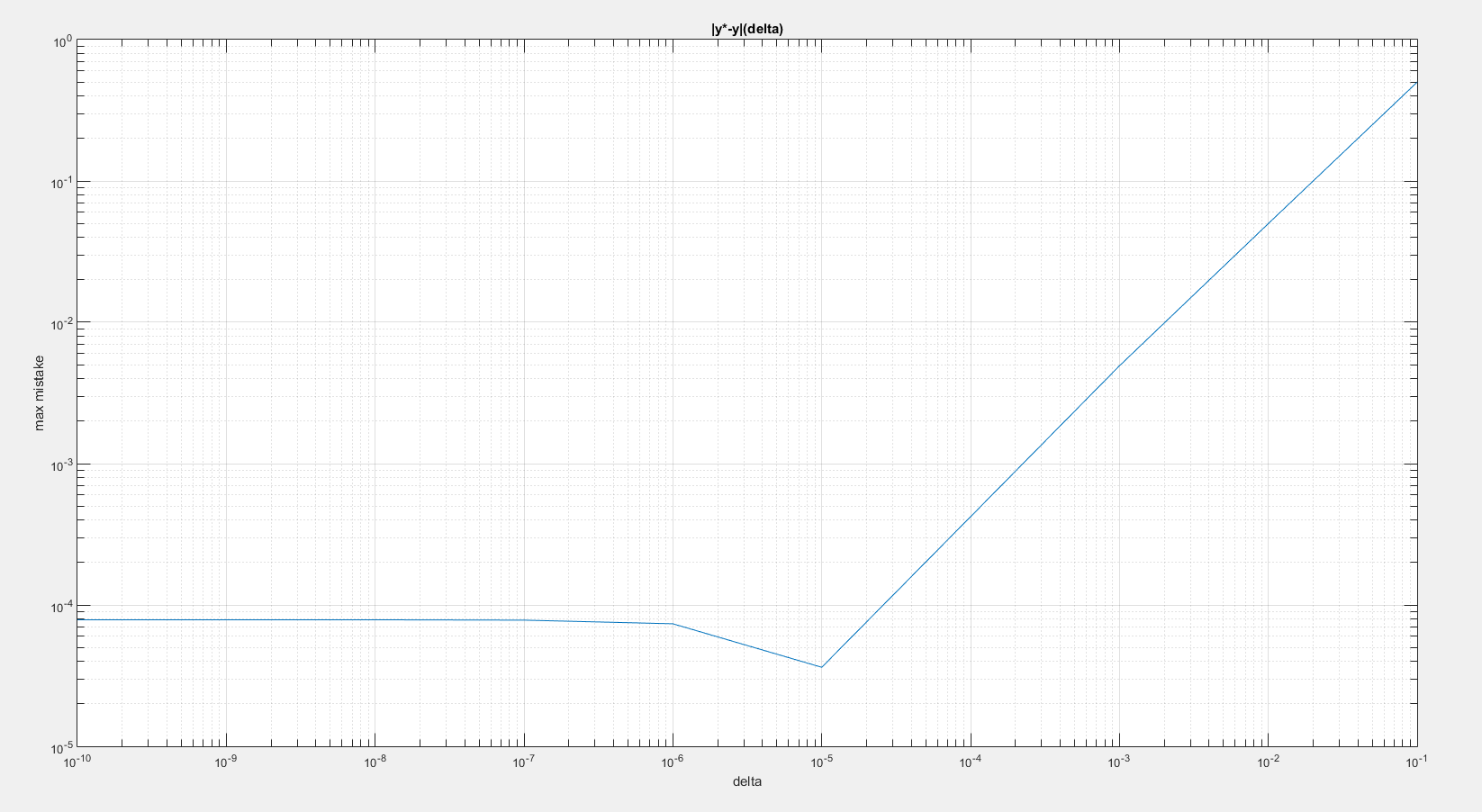


Этот график показывает практически линейную зависимость фактической точности от задаваемой. При высоких значениях наблюдается прямой участок, где фактическая точность не изменяется, а при малых точность может превышать её на целый порядок из-за накапливающейся ошибки вычислений.



Итерации от точности

Зависимость числа итераций от точности близка к линейной. При больших значениях появляется ровный участок, на котором число итераций не изменяется.



При малых значениях ошибка вычисления остаётся постоянной, однако при график сначала делает изгиб, а потом начинает заметно возрастать с линейной скоростью.

9. Вывод

Метод Эйлера-Коши оказался линейно-зависимым по всем параметрам, но значения не слишком удовлетворительные.

Фактическая точность почти на порядок ниже задаваемой, а число итераций достаточно велико, в то время как более сложные методы высших порядков могут показывать куда лучшие результаты.

Плюсом метода является линейная зависимость и простота в реализации, что делает его более выгодным для быстрых и не очень точных расчётов. Однако, для более сложных задач лучше выбрать другие методы более высоких порядков.