Cанкт – Петербургский политехнический университет Петра Великого

Физико-механический институт

Отчет по лабораторной работе №4 по Численным методам

«Численное решение задачи Коши»

Метод Рунге-Кутты 3 порядка

Выполнил студент

Группы 5030301/10002 Тугай В.В.

Преподаватель: К.Н. Козлов

Санкт-Петербург 2023

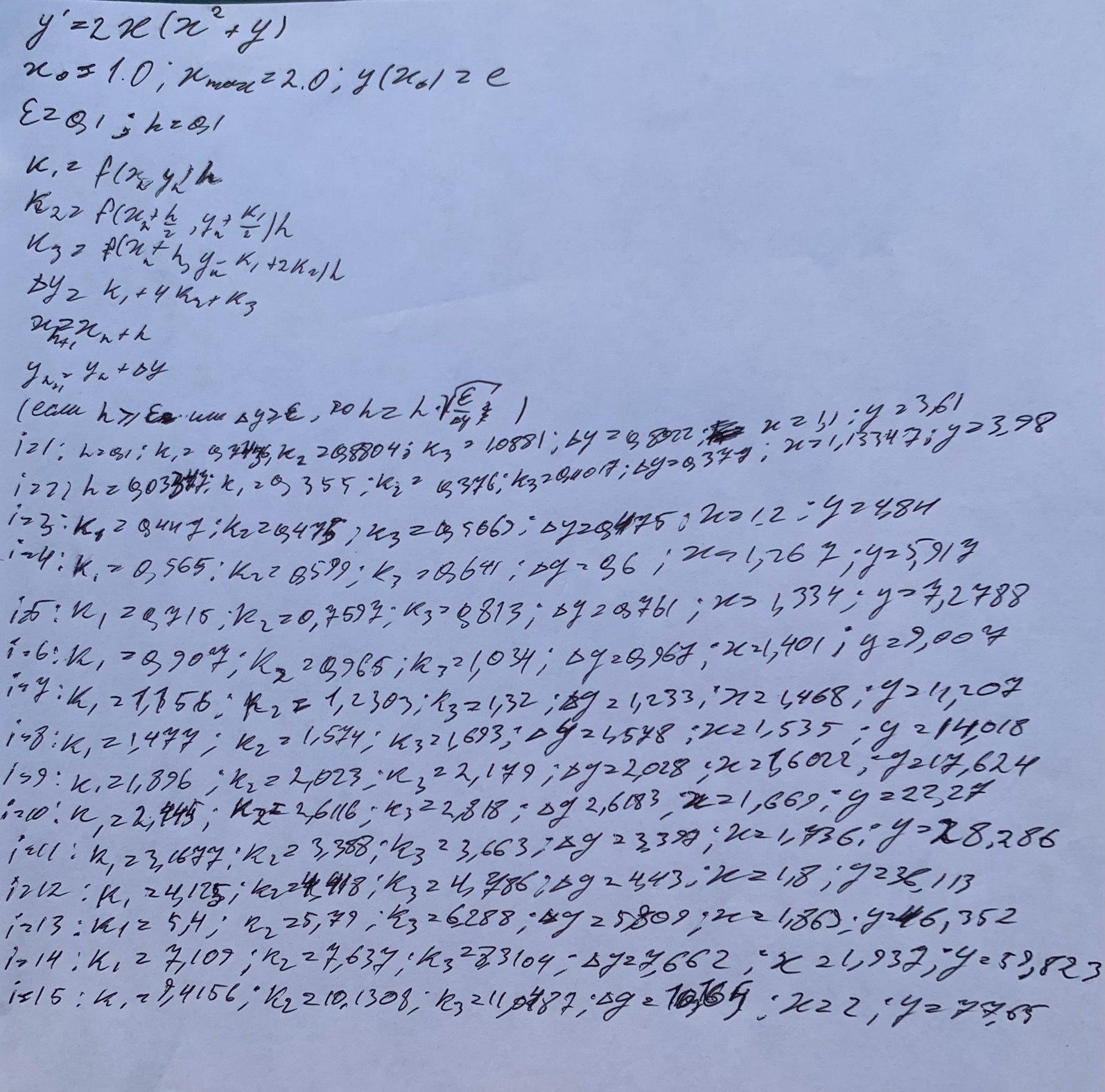
# Содержание отчета:

1. Ручной расчет
2. Метод Рунге-Кутты 3 порядка
3. Структура программы
4. Изменение фактической ошибки от шага на отрезке
5. Изменение шага на отрезке
6. Зависимость фактической от заданной точности
7. Изучение солверов MATLAB ode\*\*
8. Реализация и исследование поведения явного метода Эйлера для неустойчивой задачи
9. Реализация и исследование поведения явного и неявного метода Эйлера
10. Исследование влияния шага на решение для системы
11. Вывод

**Вариант 21**

Метод Рунге-Кутты 3 порядка

**Ручной расчет**

****

**Метод Рунге–Кутты 3 порядка**

Рассмотрим задачу Коши для системы обыкновенных дифференциальных уравнений первого порядка

Тогда приближенное значение в последующих точках вычисляется по итерационной формуле

где h – величина шага сетки по x.

**Структура программы**

Программа lab4.f95 состоит из основного тела и модуля:

RK\_Runge.mod - модуль, содержащий две функции и подпрограмму. Функция func(x,y) инициализирует исходную функцию задачи Коши. Функция Runge(a,b,eps) является алгоритмом, реализующим правило Рунге. Подпрограмма RK реализует метод Рунге–Кутты 3 порядка.

Программа lab4.m реализует численное решение задачи Коши с помощью средств Matlab. После выполнения в папке будут созданы рисунки, все они представлены далее в отчете.

**Изменение фактической ошибки от шага на отрезке**

На рисунке 1 изображены графики точного и численного решений для значений шага на отрезке и . Как можно увидеть по рисунку, численное решение задачи Коши методом Эйлера с шагом 10−2 и 10−6 совпадает с точным решением. При излишне малом значении величины шага может происходить численной нестабильности и колоссальный рост вычислительной сложности. Для предотвращения этих процессов в программе имеется параметр hmax, ограничивающий минимальное значение шага на отрезке (hmax=eps).

**A graph of a function

Description automatically generated**

Рисунок 1

График точного и численного решений

для шага на отрезке h=

**Изменение шага на отрезке**

На рисунке 2 изображен график изменения шага на отрезке. Как видно из рисунка, шаг на заданном отрезке резко уменьшается до 10−6, что связано с формулой, по которой считается длина шага h=h\*(ɛ/error)1/2.

**A graph on a white surface

Description automatically generated**

Рисунок 2

График изменения шага на отрезке

**Зависимость фактической от заданной точности**

На рисунке 3 изображен график зависимости фактической точности от заданной точности. Как можно увидеть по рисунку, метод Рунге-Кутты 3 порядка достигает заданной точности.

A graph with a purple line

Description automatically generated

Рисунок 3

Зависимость фактической от заданной точности

**Изучение солверов MATLAB ode\*\***

На рисунках 5-11 изображены зависимости фактических ошибок от параметров встроенной функции MATLAB, которые используются для решения уравнительной задачи. Некоторые функции ode требуют явной информации о свойствах якобинана. Обычно это требуется только в том случае, если уравнения являются жесткими (например, функции ode14s и ode22t. Все функции ode по умолчанию адаптируют шаг интегрирования с целью достижения заданной точности решения.

Ode автоматически подстраиваются к шагу интегрирования для достижения требуемой точности. Они используют встроенные алгоритмы для анализа поведения системы и изменения шага интегрирования. Некоторые функций ode не поддерживают обнаружение событий. Обнаружение событий позволяет остановить интегрирование, когда определенное условие или событие происходит в процессе решения дифференциального уравнения (функции ode45, ode23, ode15s, ode23s, ode23t, ode23tb).

В большинстве функций ode в Matlab не требуется явное указание свойств матрицы масс. Однако в некоторых случаях, когда система дифференциальных уравнений имеет особую структуру или требуется дополнительная настройка, можно задать свойства матрицы масс для уточнения процесса интегрирования. Например, функции ode15s и ode23t поддерживают задание свойств матрицы масс.

A graph with red dots

Description automatically generatedA graph with red dots

Description automatically generated

Рисунок 5 Рисунок 6

Зависимость фактической ошибки от Зависимость фактической ошибки от

заданной точности для ode15s заданной точности для ode23

A graph with red dots

Description automatically generatedA graph with red dots

Description automatically generated

Рисунок 7 Рисунок 8

Зависимость фактической ошибки от Зависимость фактической ошибки от

заданной точности для ode23s заданной точности для ode23t

A graph with red dotted line

Description automatically generatedA graph with numbers and a line

Description automatically generated

Рисунок 9 Рисунок 10

Зависимость фактической ошибки от Зависимость фактической ошибки от

заданной точности для ode23tb заданной точности для ode45

A graph with red dots

Description automatically generated

Рисунок 11

Зависимость фактической ошибки от заданной точности для ode113

**Реализация и исследование поведения явного метода Эйлера для неустойчивой задачи**

На рисунках 12-16 изображены зависимости для явного метода Эйлера. Как видно из рисунков, шаг интегрирования напрямую влияет на точность результата. В случае явного метода Эйлера чем меньше шаг интегрирования, тем точнее может быть получено решение.

A graph of a function

Description automatically generatedA graph with numbers and lines

Description automatically generated

Рисунок 12 Рисунок 13

Поведение неявного метода Эйлера Решение явным методом Эйлера для различных

величин шага на отрезке (0.8;5)

A graph of a number of objects

Description automatically generated with medium confidenceA graph showing a line

Description automatically generated with medium confidence

Рисунок 14 Рисунок 15

Решение неявным методом Эйлера Зависимость погрешности от

для разного шага шага на отрезке (1;2)

A graph on a white background

Description automatically generated

Рисунок 16

Зависимость погрешности от заданного шага на отрезке (0,08;5)

**Реализация и исследование поведения явного и неявного метода Эйлера**

На рисунках 17-21 изображены зависимости для невяного метода Эйлера. Как видно из рисунков, в отличие от явного метода Эйлера, точность неявного метода Эйлера обычно напрямую не зависит от шага интегрирования. Это связано с итерационным процессом, используемым для вычисления значений на следующем шаге. Неявный метод Эйлера может давать более точные результаты даже при больших шагах интегрирования, поскольку учитывает изменение значений на всем интервале шага. Кроме того, неявный метод Эйлера более устойчив, чем явный метод Эйлера.

A graph with a line

Description automatically generatedA graph of a line

Description automatically generated with medium confidence

Рисунок 17 Рисунок 18

Поведение неявного метода Эйлера Решение явным методом Эйлера для различных

величин шага на отрезке (0.8;5)

A screenshot of a graph

Description automatically generatedA graph with a line

Description automatically generated

Рисунок 19 Рисунок 20

Решение неявным методом Эйлера Зависимость погрешности от

для разного шага шага на отрезке (1;2)

A graph with a line

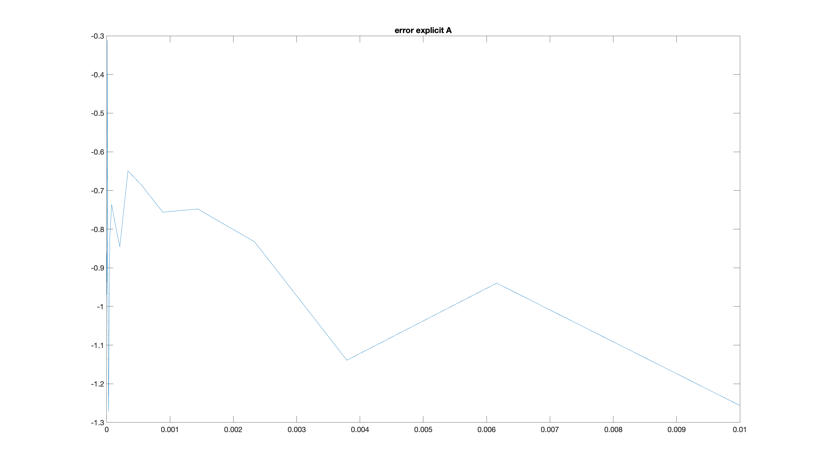
Description automatically generated

Рисунок 21

Зависимость погрешности от заданного шага на отрезке (0,08;5)

**Исследование влияния шага на решение для системы**

На рисунках 22-23 изображены зависимости ошибки от жесткости для явного и неявного методов Эйлера. Как видно из рисунков, при малой жесткости системы явный метод Эйлера может быть достаточно точным, особенно при использовании достаточно малого шага интегрирования. Ошибка метода будет относительно небольшой. Однако при повышенной жесткости системы явный метод Эйлера может давать неудовлетворительные результаты. Неявный метод Эйлера более устойчив по отношению к жестким системам. Он может успешно интегрировать системы с высокой жесткостью, где явный метод Эйлера может дать неудовлетворительные результаты.

A graph with a line

Description automatically generated

Рисунок 22 Рисунок 23

Зависимость ошибки от жесткости Зависимость ошибки от жесткости

системы, явный метод Эйлера системы, неявный метод Эйлера

**Вывод**

В ходе проделанной лабораторной работы нами был исследован метод Рунге-Кутты 3-го порядка для решения задачи Коши, а также были исследованы встроенные функции ode, явный и неявный метод Эйлера средствами пакета Matlab. Метод и алгоритм были реализованы на языке Fortran, затем по полученным результатам с помощью пакета Matlab были построены графики зависимостей. Исследования же (вместе с построением графиков) были проведены с помощью пакета Matlab.