

Аннотация

Тема бакалаврской работы «Исследование численных методов решения систем дифференциальных уравнений».

Объектом исследования являются численные методы.

Предметом данной работы являются реализованные на языке MATLAB алгоритмы метода Эйлера и метода Рунге-Кутты 4-го порядка.

В ходе работы были выполнены следующие задачи:

- рассмотрены ключевые вопросы в области численных методов;
- реализовано приложение для решения систем дифференциальных уравнений численными методами;
- проведен сравнительный анализ полученных результатов.

В первом разделе обзревается существующие численные методы решения систем дифференциальных уравнений. Приведены основные понятия в данной области, актуальность численных методов.

Во втором разделе реализуются алгоритмы численных методов решения систем дифференциальных уравнений на языке MATLAB.

В третьем разделе проводится тестирование разработанного приложения и сравнительный анализ результатов.

В итоге работы было реализовано приложение для решения систем дифференциальных уравнений численными методами, получены результаты и проведен сравнительный анализ.

Данная работа состоит из 44 страниц, включая в себя введение, три главы, заключение, 13 рисунков, 29 формул, список используемой литературы из 27 источников и приложение.

Abstract

The title of the graduation work is *Exploring the numerical methods for solving differential equations systems*.

This graduation work consists of 13 figures, 29 formulas and a list of 27 sources.

The object of the research is the numerical methods.

The study is devoted to the problems of solving a system of differential equations and providing a software implementation of an application in MATLAB programming language.

The purpose of the study is study of the numerical methods for solving differential equations systems.

To achieve the goal, the following tasks are set:

- to consider the key issues in the area of the numerical methods;
- to implement an application for solving differential equations systems by using the numerical methods;
- to carry out a comparative analysis of the obtained results.

The graduation work consists of an introduction, three chapters, a conclusion and a list of references.

The introduction explains the relevance of the study.

The first chapter of the research reviews the existing numerical methods for solving differential equations systems. It discusses the main notions and concepts of this sphere and reveals the relevance of the numerical methods.

The second chapter describes the development of an application for solving differential equations systems by using the numerical methods.

The third chapter tests the developed application and conducts a comparative analysis of the results obtained.

Содержание

Введение.....	5
1 Обзор существующих численных методов решения дифференциальных уравнений.....	8
1.1 Общие понятия систем дифференциальных уравнений и постановка задачи на исследование	8
1.2 Обзор основных численных методов решения систем дифференциальных уравнений.....	11
1.2.1 Метод Эйлера	11
1.2.2 Метод Рунге-Кутты.....	15
2 Реализация численных методов решения систем дифференциальных уравнений в системе MATLAB	19
2.1 Структура приложения.....	20
2.2 Подготовка основы программы для реализации методов	22
2.3 Реализация метода Эйлера	26
2.4 Реализация метода Рунге-Кутты 4-го порядка.....	29
3 Тестирование и анализ полученных результатов	32
3.1 Проведение эксперимента.....	32
3.2 Результаты эксперимента.....	39
Заключение	41
Список используемых источников.....	42
Приложение А Исходный код разработанного программного обеспечения.	45

Введение

«Математика как наука возникла в связи для необходимости решения практических задач: измерений на местности, навигации и т.д. Вследствие этого математика была численной математикой, ее целью было получение решения в виде числа.

Численное решение прикладных задач всегда интересовало математиков. Крупнейшие представители прошлого объединяли в своих исследованиях изучения явлений природы, получение их математического описания, как иногда говорят, математической модели явления, и его исследование. Анализ усложненных моделей потребовал создания специальных, как правило, численных методов решения задач. Названия некоторых из таких методов – методы Ньютона, Эйлера, Лобачевского, Гаусса, Зейделя, Чебышева, Эрмита, Крылова, Рунге, Кутта – свидетельствуют о том, что их разработкой занимались крупнейшие ученые своего времени» [14].

«Прогресс в развитии численных методов способствовал постоянному расширению области применения математики в других научных дисциплинах и прикладных разработках, из которых, в свою очередь, поступали запросы на решение новых задач, стимулируя дальнейшее развитие вычислительной математики. Метод математического моделирования, основанный на построении и изучении математических моделей различных объектов, процессов и явлений и получении информации о них из решения математических задач, связанных с этими моделями, стал одним из основных методов исследований в так называемых точных науках» [7].

«При математическом моделировании ряда технических устройств используются системы дифференциальных уравнений. Такие модели используются не только в технике, они находят применение в экономике, химии, биологии, медицине, управлении. Исследование функционирования подобных устройств требуют решения, обозначенных выше, систем

уравнений. Поскольку основная часть таких уравнений являются линейными и нестационарными, часто невозможно получить их аналитическое решение. Возникает необходимость использовать численные методы» [16].

Актуальность данной работы заключается в том, что системы линейных дифференциальных уравнений имеют аналитически сложное решение, и составление программы, реализующей их численное решение, облегчило бы эту задачу.

Объектом исследования в данной работе являются численные методы решения систем дифференциальных уравнений.

Предмет исследования - алгоритмы численных методов, реализованные на выбранном языке программирования.

Целью работы является исследование численных методов решения систем дифференциальных уравнений.

Для достижения поставленной цели в рамках выпускной квалификационной работы необходимо решить следующие задачи:

- провести анализ численных методов решения систем дифференциальных уравнений;
- реализовать некоторые из имеющихся численных методов;
- провести тестирование и анализ полученных результатов;
- провести сравнение выбранных численных методов исходя из полученных результатов.

Бакалаврская работа состоит из введения, трех разделов и заключения.

В первом разделе обозреваются существующие численные методы решения систем дифференциальных уравнений. Приведены основные понятия в данной области, актуальность численных методов.

Во втором разделе представлена разработка приложения решения систем дифференциальных уравнений численными методами на выбранном языке программирования.

Третий раздел посвящается тестированию приложения и сравнительному анализу результатов, полученных при помощи

реализованных алгоритмов.

1 Обзор существующих численных методов решения дифференциальных уравнений

«Численные методы это увлекательное и чрезвычайно важное направление современной математики, связанное с вычислениями на компьютере и решением сложных задач.

Чистая математика изучает движение чисел - арифметика, движение фигур – геометрия, но как числа соединяются с реальным миром? Как от формул перейти к окружающему нас миру?

Ответ на это дают численные методы, математика, соединенная с силой компьютера, позволяет заглянуть вглубь реального мира, промоделировать сложнейшие физические, технические и биологические процессы.

Наблюдая процессы реального мира, мы вначале описываем их вербально, пытаюсь понять суть явлений, далее строим математические модели.

Однако, мы не хотим ограничиваться построением формальных моделей, а хотим получить качественное и количественное представление об изучаемых процессах, увидеть их на графиках.

Именно здесь нам на помощь приходят компьютеры и численные методы» [5].

В разделе рассматриваются различные численные методы решения систем дифференциальных уравнений, даются общие представления и терминология. Приводятся общие понятия из теории и ставится задача на исследование.

1.1 Общие понятия систем дифференциальных уравнений и постановка задачи на исследование

Дифференциальным уравнением называется уравнение, которое содержит производные неизвестной функции (или нескольких неизвестных функций). Вместо производных могут содержаться дифференциалы.

«Чтобы решить обыкновенное дифференциальное уравнение, необходимо знать значения зависимой переменной и (или) её производных при некоторых значениях независимой переменной».

Дифференциальным уравнением (ДУ) n-го порядка является пропорция следующего вида:

$$H(t, x, x', x'', \dots, x^{(n)}) = 0 \quad (1)$$

Решением ДУ является функция $x(t)$, превращающая уравнение в равенство, справедливое при любых числовых значениях, т.е. тождество.

Системой дифференциальных уравнений первого порядка является комплекс ДУ первого порядка, содержащих независимую переменную, неизвестные функции и их производные.

Системы дифференциальных уравнений бывают двух основных типов - линейные однородные и неоднородные.

«Системой дифференциальных уравнений n-го порядка называется система вида:

$$\begin{cases} x'_1 = f_1(t, x_1, x_2, \dots, x_n) \\ x'_2 = f_2(t, x_1, x_2, \dots, x_n) \\ \dots \dots \dots \dots \dots \\ x'_n = f_n(t, x_1, x_2, \dots, x_n) \end{cases} \quad (2)$$

Системой линейных дифференциальных уравнений называется система вида:

$$\begin{cases} x'_1 = \sum_{j=1}^n a_{1j}x_j + b_1 \\ x'_2 = \sum_{j=1}^n a_{2j}x_j + b_2 \\ \dots \dots \dots \dots \dots \\ x'_n = \sum_{j=1}^n a_{nj}x_j + b_n \end{cases} \quad (3)$$

Решением системы называется вектор

$$x(t) = \begin{cases} x_1(t) \\ x_2(t) \\ \dots \dots \\ x_n(t) \end{cases} \quad (4)$$

который обращает уравнения (2) и (3) в тождества.

«Каждое дифференциальное уравнение, так же, как и система, имеет бесконечное множество решений, которые отличаются друг от друга константами. Для однозначного определения решения необходимо определить дополнительные начальные или граничные условия. Количество таких условий должно совпадать с порядком дифференциального уравнения или системы. В зависимости от вида дополнительных условий в дифференциальных уравнениях различают:

- задачу Коши, в случае если все дополнительные условия заданы в одной (чаще начальной) точке интервала;
- краевую задачу, в случае, когда дополнительные условия заданы на границах интервала.

Различают точные (аналитические) и численные (приближенные) методы решения дифференциальных уравнений. Большое количество уравнений может быть решено точно. Однако есть уравнения, а особенно системы уравнений, для которых нельзя записать точное решение. Но даже для уравнений с известным аналитическим решением очень часто необходимо вычислить числовое значение при определенных исходных данных. Поэтому широкое распространение получили численные методы решения обыкновенных дифференциальных уравнений» [6].

В исследовании, производимом в данной бакалаврской работе, будут использоваться линейные системы дифференциальных уравнений. Так же в этой работе будут разработаны алгоритмы численных методов решения систем дифференциальных уравнений на выбранном языке программирования, и будут исследованы на предмет точности выбранных методов.

1.2 Обзор основных численных методов решения систем дифференциальных уравнений

«В классическом математическом анализе разработано достаточно приемов нахождения решений дифференциальных уравнений через элементарные функции. Между тем при решении практических задач эти методы оказываются, как правило, либо совсем бесполезными, либо их решение связано с недопустимыми затратами усилий и времени. Для решения прикладных задач созданы методы приближенного решения дифференциальных уравнений, которые условно можно подразделить на три основные группы:

- Аналитические методы, применение которых даст решение ОДУ в виде аналитической функции;
- Графические методы, дающие приближенное решение в виде графика;
- Численные методы, когда искомая функция получается в виде таблицы» [2].

В работе будет рассматриваться последняя группа.

1.2.1 Метод Эйлера

Метод Эйлера впервые был описан швейцарским математиком Леонардом Эйлером в 1768 году в его работе «Интегральное исчисление». Он считается наиболее простым численным методом решения дифференциальных уравнений. Еще о методе Эйлера можно сказать, что это явный, одношаговый метод первого порядка точности.

«Это один из старейших и наиболее известных методов. Метод Эйлера является относительно грубым методом решения дифференциальных уравнений, однако идеи, лежащие в его основе, по сути, являются исходными для очень широкого класса численных методов.

Он основан на идее графического построения решения дифференциального уравнения, но этот метод одновременно предоставляет способ найти искомую функцию в числовой форме или в таблице».

«Метод Эйлера играет важную роль в теории численных методов решения ОДУ хотя и не часто используется в практических расчетах из-за невысокой точности. Вывод расчетных соотношений для этого метода может быть произведен несколькими способами: с помощью геометрической интерпретации с использованием разложения в ряд Тейлора конечно разностным методом с помощью разностной аппроксимации производной квадратурным способом использованием эквивалентного интегрального уравнения» [20].

Рассмотрим уравнение:

$$y' = f(x, y) \quad (5)$$

в окрестностях узлов $x = x_i, (i = 0, 1, \dots)$ и заменим в левой части производную y' правой разностью. При этом значения функции y в узлах x_i заменим значениями сеточной функции $y(x_i) = y_i$:

$$\frac{y_{i+1} - y_i}{h_i} = f(x_i, y_i). \quad (6)$$

Говорят, что уравнение (7) аппроксимирует исходное уравнение (6) с первым порядком, так как погрешность аппроксимации определяется, как $O(h_i)$.

Рассмотрим равномерную сетку, с узлами, равноотстоящими друг от друга, $h_i = x_{i+1} - x_i = h = const, (i = 0, 1, \dots)$. Тогда из равенства (7) получаем:

$$y_{i+1} = y_i + hf(x_i, y_i), i = 0, 1, \dots \quad (7)$$

Заметим, что из уравнения (1.5) при $h \rightarrow 0$ следует

$$y'(x_i) = f(x_i, y(x_i)) = f(x_i, y_i). \quad (8)$$

Уравнение (7) позволяет приблизительно определить значение функции y в точке x_{i+1} используя разложения в ряд Тейлора с отбрасыванием членов второго и более высоких порядков. Другими словами, предполагается, что приращение функции равно ее дифференциалу.

Полагая $i = 0$, с помощью соотношения (7) можно определить значение сеточной функции y при $x = x_1, y_1$:

$$\delta_1 = O(h^2), \delta_2 = \delta_1 + O(h^2) + hO(h^2) = O(h^2)(2 + h) \approx O(h^2) \quad (15)$$

Здесь мы видим, что последнее слагаемое в (11) можно отбросить. Уравнение примет вид:

$$\delta_{i+1} = \delta_i + O(h^2) \quad (16)$$

т. е. погрешность на каждом шаге увеличивается на величину $O(h^2)$.

При нахождении решения в точке x_n , отстоящей на конечном расстоянии L от точки x_0 , погрешность состоит из n слагаемых $O(h^2)$. Если учесть, что $h=L/n$, то для погрешности δ_n получаем окончательное выражение:

$$\delta_n = nO(h^2) = \frac{L}{h}O(h^2) = O(h) \quad (17)$$

Получается, что метод Эйлера имеет первый порядок точности».

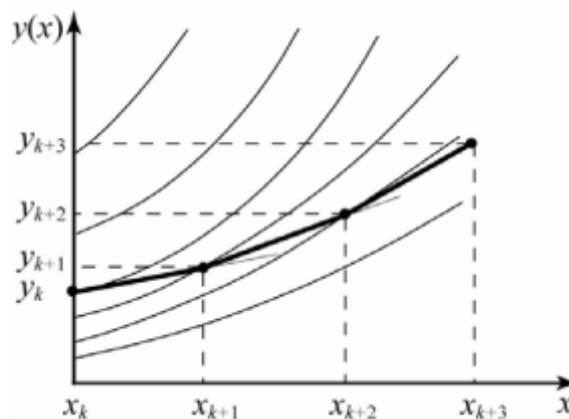


Рисунок 1 – Геометрическая интерпретация метода Эйлера

«Рассмотренный метод Эйлера (7) является частным случаем методов первого и второго порядков, относящихся к классу методов Рунге-Кутты. Эти методы применяют для вычисления значения y_{i+1} , ($i = 0, 1, \dots$) через y_i и $f(x, y)$, определенных при некотором специальном образом выбираемых

значениях $x \in [x_i, x_{i+1}]$ и $y(x)$. На их основе могут быть построены разностные схемы разного порядка точности. Одним из наиболее часто используемых методов является метод Рунге-Кутты четвертого порядка. Алгоритм метода записывается в виде:

$$\left\{ \begin{array}{l} y_{i+1} = y_i + \frac{1}{6}(k_0 + 2k_1 + 2k_2 + k_3), i = 0, 1, \dots \\ k_0 = hf(x_i, y_i) \\ k_1 = hf\left(x_i + \frac{h}{2}, y_i + \frac{k_0}{2}\right) \\ k_2 = hf\left(x_i + \frac{h}{2}, y_i + \frac{k_1}{2}\right) \\ k_3 = hf(x_i + h, y_i + k_2) \end{array} \right. \quad (18)$$

Данный метод требует на каждом шаге четырехкратного вычисления правой части $f(x, y)$ уравнения (7). Суммарная погрешность этого метода есть величина $O(h^4)$ » [19].

1.2.2 Метод Рунге-Кутты

Методы Рунге-Кутты были открыты в 1900 году двумя немецкими математиками Карлом Давидом Тольме Рунге и Мартином Вильгельмом Куттой. Это весьма объемный класс численных методов решения задачи Коши для ОДУ и их систем.

«Метод Рунге-Кутта требует большего объема вычислений по сравнению с методом Эйлера, однако это окупается повышенной точностью, что дает возможность проводить счет с большим шагом. Другими словами, для получения результатов с одинаковой точностью в методе Эйлера потребуется значительно меньший шаг, чем в методе Рунге-Кутта» [19].

«Рассмотрим задачу Коши, где правая часть удовлетворяет условиям теорем существования и единственности решения.

$$y' = f(x, y), a \leq x \leq b; y(a) = y^0 \quad (19)$$

Зададим равномерную сетку:

$$x_i = a + ih, i = 1, \dots, n, h = \frac{b - a}{n} \quad (20)$$

Введем обозначения $y(x_i) = y_i$. Получим вычислительную формулу:

$$\begin{cases} k_1 = hf(x_i, y_i) \\ k_2 = hf(x_i + h/2, y_i + k_1/2) \\ k_3 = hf(x_i + h/2, y_i + k_2/2) \\ k_4 = hf(x_i + h, y_i + k_3) \\ y_{i+1} = y_i + [k_1 + 2k_2 + 2k_3 + k_4]/6 \end{cases} \quad (21)$$

«Численное решение задачи Коши для систем ОДУ 1-го порядка методами Рунге-Кутты ищется по тем же формулам, что и для ОДУ первого порядка». Например, решение методом Рунге-Кутты 4-го порядка можно найти, если положить:

$$\begin{aligned} y_i &\rightarrow \bar{y}_i \\ f(x_i, y_i) &\rightarrow \bar{f}(x_i, \bar{y}_i) \\ k_l &\rightarrow \bar{k}_l \\ \bar{k}_l &= \begin{pmatrix} k_{i,1}^l \\ \vdots \\ k_{i,m}^l \end{pmatrix} \end{aligned} \quad (22)$$

где m – размерность системы, $l=1, \dots, 4$. В результате получим:

$$\begin{cases} \bar{k}_1 = h\bar{f}(x_i, \bar{y}_i) \\ \bar{k}_2 = h\bar{f}(x_i + h/2, \bar{y}_i + \bar{k}_1/2) \\ \bar{k}_3 = h\bar{f}(x_i + h/2, \bar{y}_i + \bar{k}_2/2) \\ \bar{k}_4 = h\bar{f}(x_i + h, \bar{y}_i + \bar{k}_3) \\ \bar{y}_{i+1} = \bar{y}_i + [\bar{k}_1 + 2\bar{k}_2 + 2\bar{k}_3 + \bar{k}_4]/6 \end{cases} \quad (23)$$

«Дополнительного повышения точности расчетов можно добиться, повысив порядок метода за счет увеличения количества операций на один шаг разностной сетки. Алгоритм расчета уравнения для метода Рунге-Кутты-Мерсона 5-го порядка представлен уравнениями».

$$y_{i+1} = y_1 + \frac{1}{6}(k_0 + 4k_3 + k_4), i = 0, 1, \dots, \quad (24)$$

$$\left\{ \begin{array}{l} k_0 = hf(x_i, y_i), \\ k_1 = hf\left(x_i + \frac{h}{3}, y_i + \frac{k_0}{3}\right) \\ k_2 = hf\left(x_i + \frac{h}{3}, y_i + \frac{k_0}{6} + \frac{k_1}{6}\right) \\ k_3 = hf\left(x_i + \frac{h}{2}, y_i + \frac{k_0}{8} + \frac{3k_2}{8}\right) \\ k_3 = hf\left(x_i + \frac{h}{2}, y_i + \frac{k_0}{8} + \frac{3k_2}{8}\right) \\ k_4 = hf\left(x_i + h, y_i + \frac{k_0}{2} - \frac{3k_2}{2} + 2k_3\right) \end{array} \right. \quad (25)$$

«Суммарная погрешность метода равна $O(h^5)$.

В случае длительных расчетов, требующих большого количества вычислений, можно сократить время расчета за счет использования переменного шага разностной сетки h . При использовании переменного шага в расчетах контролируется разность между соседними значениями сеточной функции» $\Delta = (y_i - y_{i+1})$. В случае превышения Δ заданной погрешности ε шаг сетки уменьшается в два раза, при малых значениях Δ шаг увеличивается в два раза. Условия автоматического выбора шага сетки представлены уравнениями» (26).

$$h_{i+1} = \begin{cases} 2 \times h_i, \Delta \leq \frac{5}{32} \varepsilon; \\ 0.5 \times h_i, \Delta \geq 5\varepsilon; \\ h_i, \frac{5}{32} \varepsilon < \Delta < \varepsilon \times 5. \end{cases} \quad (26)$$

«Условия (26) позволяют существенно сократить время расчета задачи, сохранив точность решения».

1.3 Выводы и заключения по разделу 1

В данном разделе определены основные необходимые понятия из теории дифференциальных уравнений и их численных решений. Кроме того, определена задача на исследование численных методов и представлены существующие подходы для ее решения. Также описаны использующиеся в работе алгоритмы численных методов Эйлера и Рунге-Кутты 4-го порядка. Далее показывается, как подобный алгоритм может быть обобщен для решения систем дифференциальных уравнений. В заключении этого раздела обоснован выбор численных методов, описаны их основные преимущества.

2 Реализация численных методов решения систем дифференциальных уравнений в системе MATLAB

«Под численными методами в широком смысле можно понимать интерпретацию математической модели на языке, доступном компьютеру.

Например, если математическая модель представлена в виде дифференциального уравнения, то численным методом может быть разностное уравнение, приближающее исходное дифференциальное.

Для того чтобы использовать компьютер, мы должны составить программу, реализующую данный численный метод.

Самое замечательное то, что используя компьютер, мы находим свойства процесса, о которых ранее могли только догадываться» [5].

Для реализации численных методов решения систем дифференциальных уравнений в работе будет использоваться язык MATLAB. Далее последует обоснование выбора языка для реализации приложения.

«Система MATLAB предлагается разработчиками (корпорация The MathWorks, Inc.) как лидирующий на рынке, в первую очередь на предприятиях военно-промышленного комплекса, в энергетике, в аэрокосмической отрасли и в автомобиле-строении язык программирования высокого уровня для технических вычислений, расширяемый большим числом пакетов прикладных программ – расширений. Самым известным из них стало расширение Simulink, обеспечивающее блочное имитационное моделирование различных систем и устройств. Но и без пакетов расширения MATLAB представляет собой мощную операционную среду для выполнения огромного числа математических и научно-технических расчетов и вычислений и создания пользователями своих пакетов расширения и библиотек процедур и функций. Новые версии системы имеют встроенный компилятор и позволяют создавать исполняемые файлы» [3].

«В MATLAB удачно реализованы средства работы с многомерными массивами, большими и разреженными матрицами и многими типами данных. Система прошла многолетний путь развития от узко специализированного матричного про-граммного модуля, используемого только на больших ЭВМ, до универсальной интегрированной СКМ, ориентированной на массовые персональные компьютеры класса IBM PC, AT и Macintosh, рабочие станции UNIX и даже суперкомпьютеры. MATLAB имеет мощные средства диалога, графики и комплексной визуализации вычислений» [3].

В данном исследовании выбор пал на MATLAB по причине того, что он имеет все необходимые инструменты нужные нам для реализации приложения, такие как: простой вывод графиков, возможность простого их сравнения и с помощью которых, можно облегчить само исследование, а также он позволяет при работе применять обычный математический язык, что позволяет более наглядно сравнивать реализуемые численные методы между собой.

К плюсам данного программного обеспечения я отношу: удобный пользовательский интерфейс, возможность ввода собственных дифференциальных уравнений, ну и весьма высокую стабильность работы.

Но также я выделил и ряд недостатков: критичность к вводимым данным, отсутствующая обработка исключительных событий. Эти вещи довольно сильно ограничивают варианты использования MATLAB.

2.1 Структура приложения

Проектирование приложения начинается с выделения его логических элементов, сопоставления их с физическими элементами и определения схемы их взаимодействия между собой.

«Этот этап необходим для снижения сложности при разработке, выделение подсистем и правил их взаимодействия между собой помогает избежать затрат на изменение системы, которые могут возникнуть при игнорировании этапа проектирования. При правильном проектировании на

этом этапе и следовании его результатам, в процессе разработки классов можно добиться слабого связывания самих классов и компонентов приложения в целом. Все это приводит к повышению простоты сопровождения, расширяемости и возможности повторного использования частей программы» [11].

Диаграмма компонентов разрабатываемого приложения показана на рисунке 2.

«Перед разработкой необходимо произвести анализ, позволяющий выявить части приложения, которые могут быть в последствии изменены для введения дополнительного функционала и для оптимизации производительности. На основе этой информации принимаются решения о введении интерфейсов, перегруженных методах и алгоритмов работы» [11].

Разработанная программа будет состоять из следующих функциональных элементов:

- тело программы – отвечает переменные, обращение к функции системы ОДУ и вызов функции меню для выбора метода;
- функция системы ОДУ;
- функция меню – вызывает интерфейс, из которого можно выбрать численный метод;
- функция алгоритма численного метода Рунге-Кутты 4-го порядка;
- функция алгоритма численного метода Эйлера;

Для наиболее наглядного представления общей структуры приложения разработана диаграмма компонентов, которая позволяет определить архитектуру разрабатываемой системы. В ней можно увидеть все имеющиеся компоненты приложения. Пунктирные стрелки, соединяющие модули, показывают отношения взаимозависимости.

На рисунке 1 представлена диаграмма компонентов приложения.

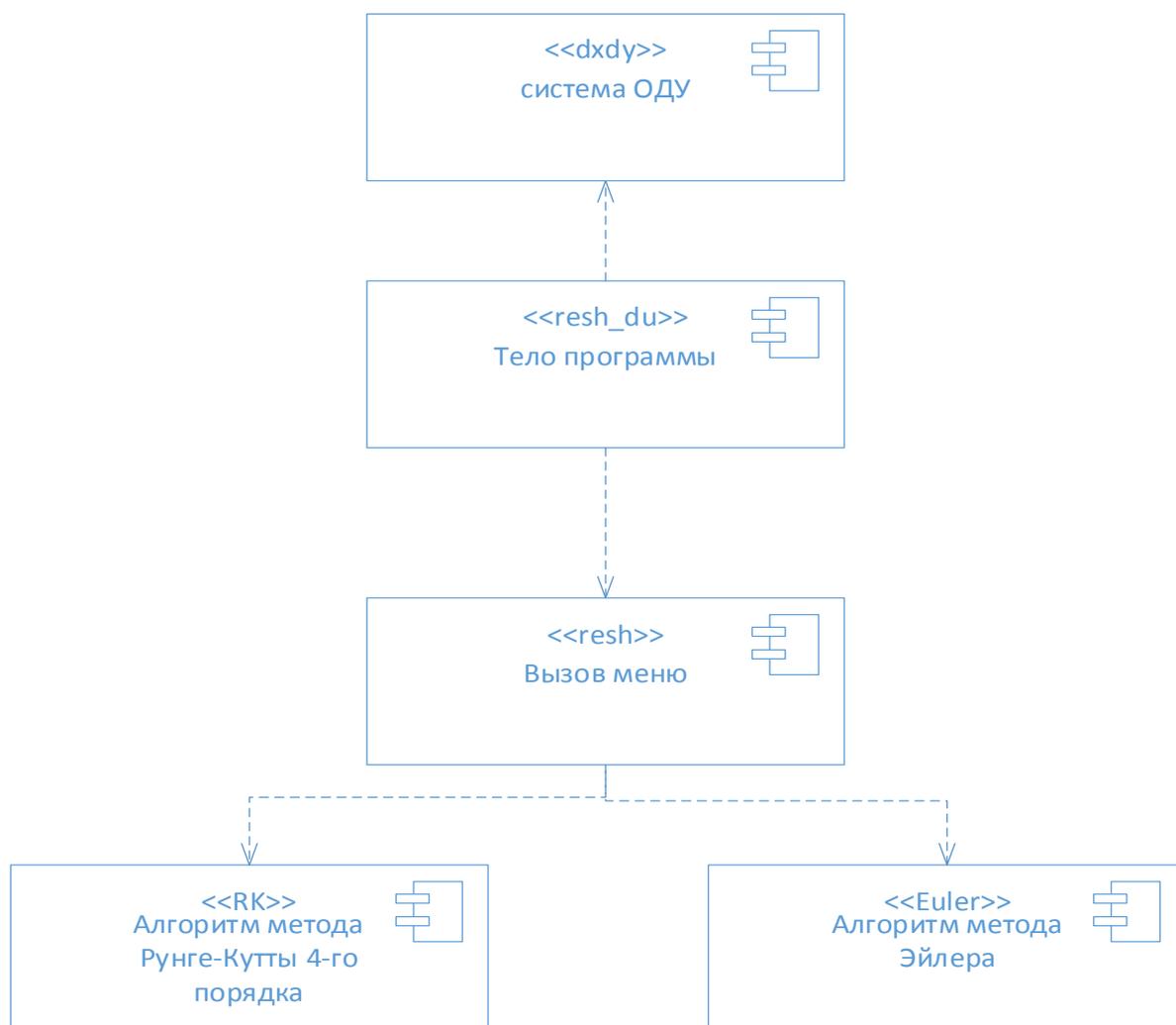


Рисунок 2 – Диаграмма компонентов программы

При написании любой программы необходимо четко представлять, что такое алгоритм, как он работает и как его реализовывать.

Также алгоритм показывает структуру выполнения программы или части кода, что очень важно при внесении изменений в разработанное приложение.

2.2 Подготовка основы программы для реализации методов

В ходе разработки программы были построены блок-схемы алгоритма для данной программы, что упростило процесс проектирования и облегчило понимание исходного кода готовой программы.

Каждый функциональный элемент будет в виде отдельного файла функции.

Первым шагом необходимо определить функцию, в которой будет задаваться система ОДУ, которую реализованные алгоритмы численных методов будут решать. На рисунке 3 показана блок-схема функции, определяющей систему обыкновенных дифференциальных уравнений.

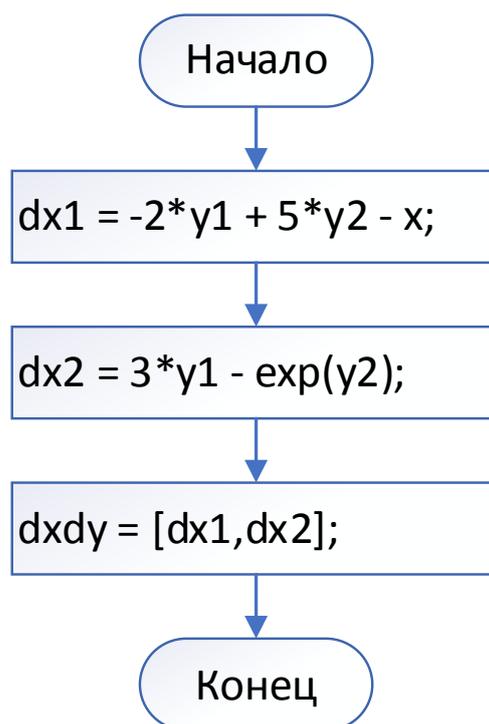


Рисунок 3 – Блок-схема функции, определяющей систему ОДУ

Данная функция состоит из двух уравнений и массива, в который будут входить оба этих уравнения.

Следующим шагом будет реализация тела приложения, в котором будут задаваться исходные данные, такие как: уравнения системы, границы интегрирования, начальные условия и шаг. После чего будет вызываться функция «меню», из которого можно будет выбрать метод решения системы ОДУ. Тело приложения показано на рисунке 4.

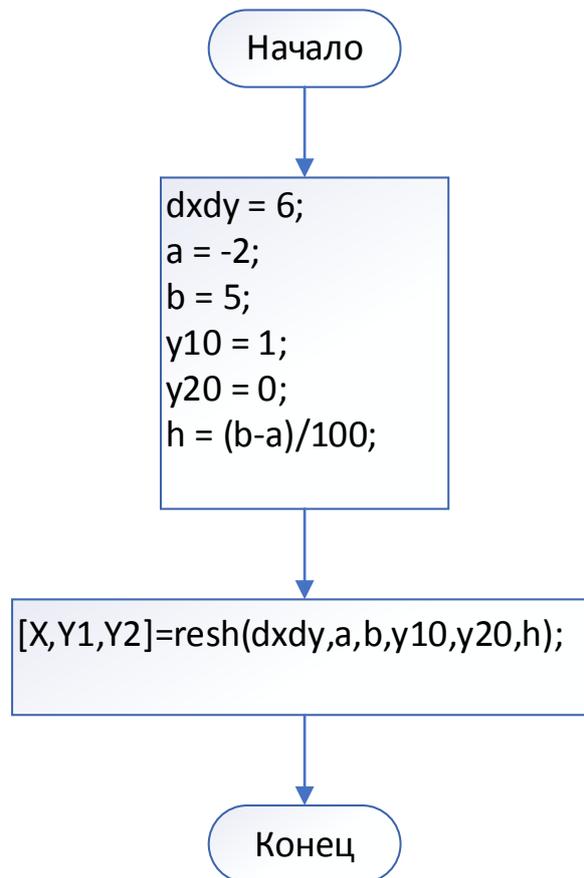


Рисунок 4 – Блок-схема тела программы

Поэтапное создание графического интерфейса приложений и сложных систем экономит время, структурирует всю работу, уменьшает вероятность дополнительных денежных вложений. Думаю, из всего вышенаписанного уже понятно, что интерфейс в принципе необходим для удобного взаимодействия пользователя с приложением. Но что самое главное — это умение разработать интерфейс, в котором пользователь найдёт ключевые функции продукта за минимально необходимое для этого время.

Далее следует реализация функции меню, из которого можно выбрать необходимые метод решения системы ОДУ. Блок-схема этой функции представлена на рисунке 5.

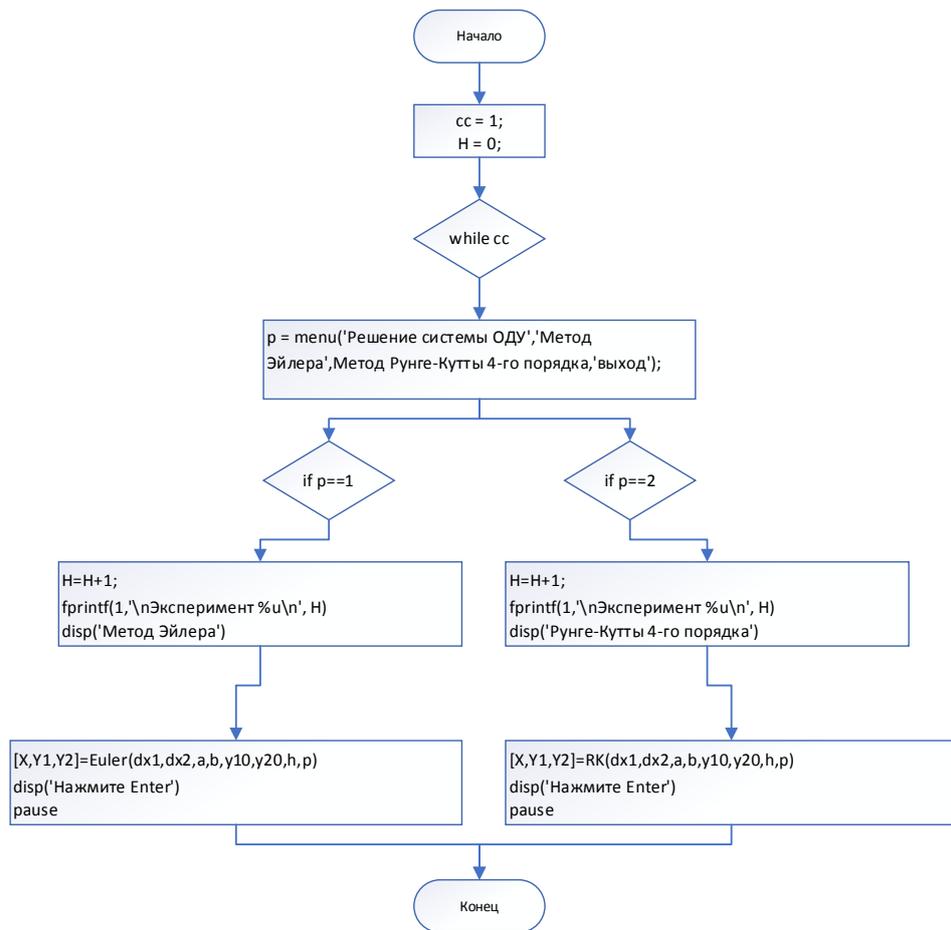


Рисунок 5 – Блок-схема функции меню приложения

Функция получает исходные данные из тела приложения, после чего, в зависимости от выбора пользователя, направляет их в один из алгоритмов реализованных численных методов.

Меню выглядит следующим образом:

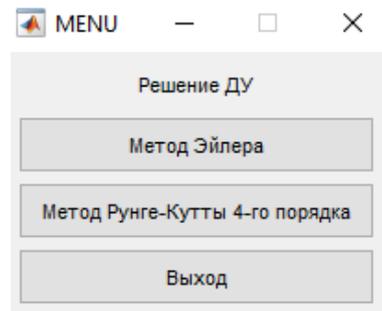


Рисунок 6 – Меню приложения.

2.3 Реализация метода Эйлера

Данная функция программы отвечает за реализацию алгоритма решения системы дифференциальных уравнений методом Эйлера. В этом подразделе будет более подробно описана суть работы алгоритма с подробной блок-схемой и описанием каждого элемента. Блок схема состоит из пяти блоков, описание того, за что отвечает каждый из них приведено ниже.

Блок 1 отвечает за принятие данных, которые перенаправляет сюда функция меню $dx1, dx2, a, b, y10, y20, h$. Переменные $dx1, dx2$ это уравнения заданной системы ОДУ; $[a, b]$ интервал, на котором ищется решение системы; $y10, y20$ первое и второе начальные условия; h – оптимальный шаг численного дифференцирования.

Блок 2 – расчет первых 2-х точек заданной сетки методом Эйлера. Аргументами функции являются: строки, которые уже были преобразованы в обратную польскую нотацию, которые задают уравнения системы; текущее значение x ; значения требуемых функций на предыдущем шаге (условно обозначенные).

Блок 3 – уменьшение шага вдвое для уменьшения погрешности.

Блок 4 – расчет следующих двух точек заданной сетки методом Эйлера.

Блок 5 – вывод решения и графика.

Для демонстрации алгоритма разработана подробная блок-схема, которая изображена на рисунке 7.

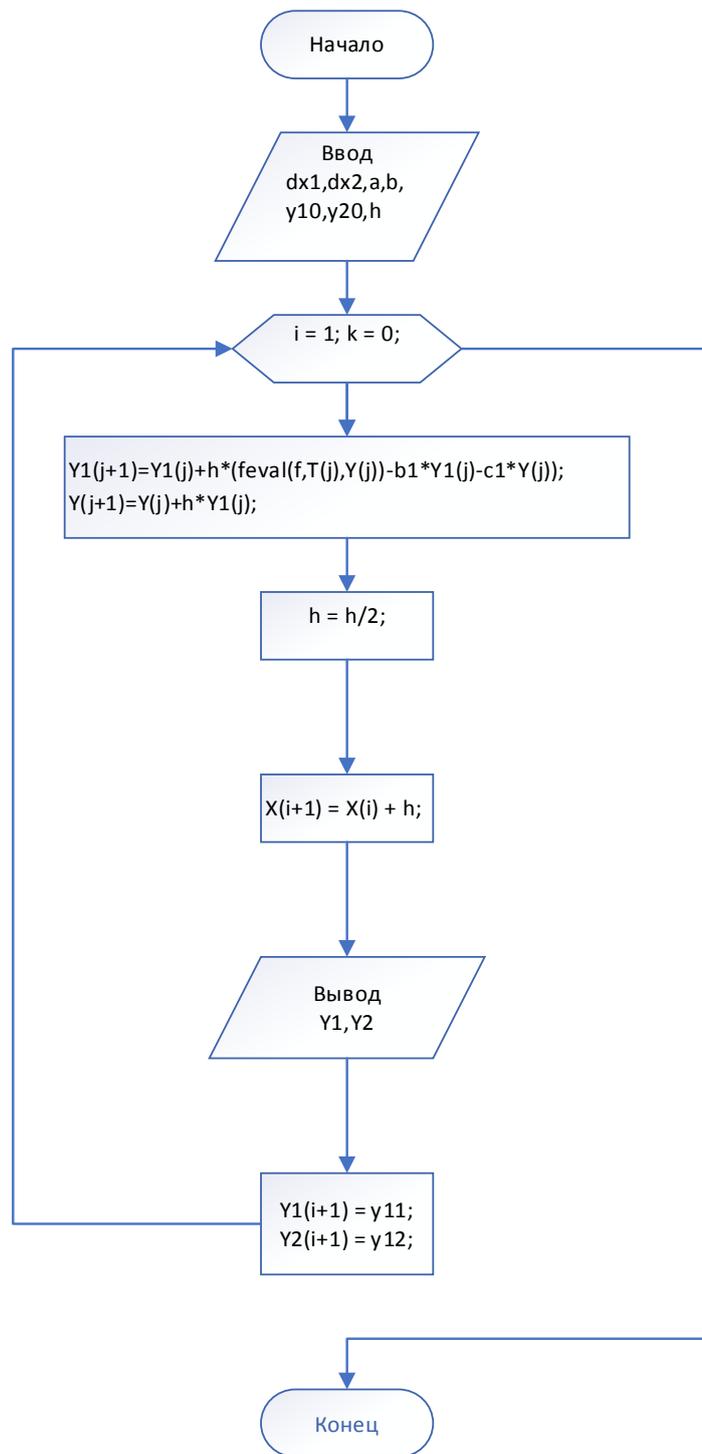


Рисунок 7 – Блок-схема алгоритма метода Эйлера

«Данная программа решает заданную пользователем систему дифференциальных уравнений с указанной точностью за минимальный промежуток времени. При этом пользователю предоставляется возможность

визуально оценить неточность решения, сравнивая графики полученного и точного решений» [13].

«К достоинствам программы можно отнести также удобный пользовательский интерфейс, возможность ввода пользовательских систем дифференциальных уравнений, а также высокая стабильность работы. Однако имеются и некоторые недостатки. К недостаткам программы можно отнести: критичность к вводимым пользователем функций, отсутствие обработки исключительных событий. Это, естественно, ограничивает возможности программы» [13].

Само же решение уравнений с помощью этой программы будет рассмотрено в разделе 3.

2.4 Реализация метода Рунге-Кутты 4-го порядка

Данная функция программы отвечает за реализацию алгоритма решения системы дифференциальных уравнений методом Рунге-Кутты 4-го порядка. В этом подразделе будет подробно описана суть работы алгоритма с подробной блок-схемой и описанием каждого элемента. Блок схема состоит из пяти блоков, описание того, за что отвечает каждый из них приведено ниже.

Данная функция программы отвечает за реализацию решения системы дифференциальных уравнений методом Рунге-Кутты 4-го порядка.

Блок 1 отвечает за принятие данных, которые перенаправляет сюда функция меню $dx1, dx2, a, b, y10, y20, h$. Переменные $dx1, dx2$ это уравнения заданной системы ОДУ; $[a, b]$ интервал на котором ищется решение системы; $y10, y20$ первое и второе начальные условия; h – оптимальный шаг численного дифференцирования.

Блок 2 - вычисление первых двух точек заданной сетки методом Рунге-Кутты 4-го порядка. Аргументами функции являются: строки, которые уже были преобразованы в обратную польскую нотацию, которые задают уравнения системы; текущее значение x ; значения требуемых функций на предыдущем шаге (условно обозначенные).

Блок 3 – уменьшение шага вдвое.

Блок 4 – расчет следующих двух точек заданно сетки методом Рунге-Кутта 4-го порядка.

Блок 5 – вывод решения и графика.

Также для демонстрации алгоритма разработана подробная блок-схема, которая изображена на рисунке 8.

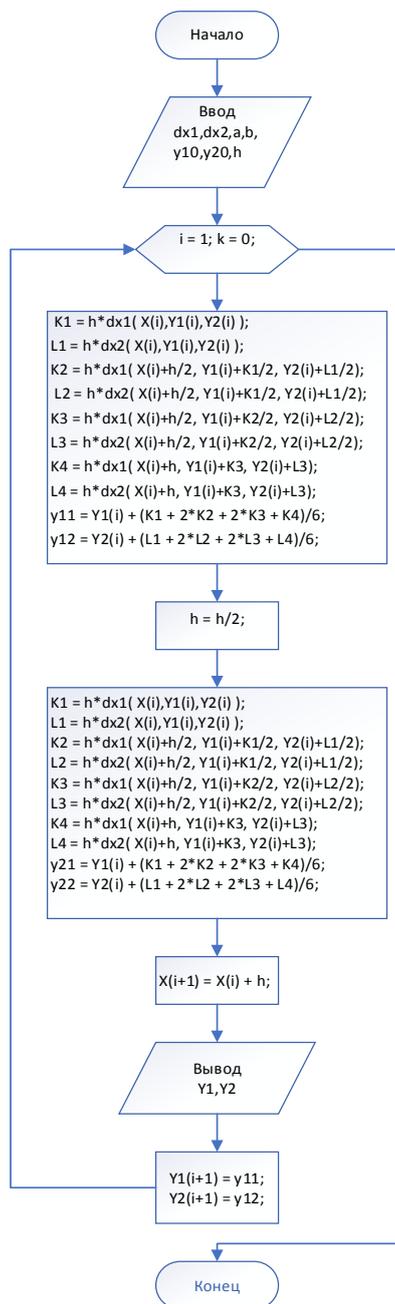


Рисунок 8 – Блок-схема алгоритма метода Рунге-Кутты 4-го порядка

Данное приложение решает заданную пользователем систему дифференциальных уравнений методом Рунге-Кутты 4-го порядка за кратчайшие сроки. Также пользователь имеет возможность визуально оценить решение, рассмотрев построенный график полученного решения.

Само же решение уравнений с помощью, обозначенной выше программы, будет рассмотрено в разделе 3.

2.5 Выводы и заключения по разделу 2

В данном разделе было реализовано на языке MATLAB приложение для решения систем дифференциальных уравнений численными методами.

Описана структура разработанного приложения, разработаны выбранные в первом разделе численные методы решения систем дифференциальных уравнений. «И легко понять, насколько это просто в этой системе. И вам не нужно быть продвинутым пользователем ПК, чтобы понять принцип работы системы MATLAB и освоить ее».

Также приведена структура разработанного приложения и блок-схемы компонентов. Описан принцип работы разработанных алгоритмов.

3 Тестирование и анализ полученных результатов

3.1 Проведение эксперимента

«Вычислительным экспериментом называется методология и технология исследований, основанные на применении прикладной математики и ЭВМ как технической базы при использовании ММ. Вычислительный эксперимент основывается на создании ММ изучаемых объектов, которые формируются с помощью некоторой особой математической структуры, способной отражать свойства объекта, проявляемые им в различных экспериментальных условиях, и включает в себя следующие этапы» [4].

1. «Для исследуемого объекта строится модель, обычно сначала физическая, фиксирующая разделение всех действующих в рассматриваемом явлении факторов на главные и второстепенные, которые на данном этапе исследования отбрасываются; одновременно формулируются допущения и условия применимости модели, границы, в которых будут справедливы полученные результаты; модель записывается в математических терминах, как правило, в виде дифференциальных или интегро-дифференциальных уравнений; создание ММ проводится специалистами, хорошо знающими данную область естествознания или техники, а также математиками, представляющими себе возможности решения математической задачи» [4].

2. «Разрабатывается метод решения сформулированной математической задачи. Эта задача представляется в виде совокупности алгебраических формул, по которым должны вестись вычисления и условия, показывающие последовательность применения этих формул; набор этих формул и условий носит название вычислительного алгоритма. Вычислительный эксперимент имеет многовариантный характер, так как решения поставленных задач часто зависят от многочисленных входных параметров. Тем не менее, каждый конкретный расчет в вычислительном эксперименте проводится при фиксированных значениях всех параметров.

Между тем в результате такого эксперимента часто ставится задача определения оптимального набора параметров. Поэтому при создании оптимальной установки приходится проводить большое число расчетов однотипных вариантов задачи, отличающихся значением некоторых параметров. В связи с этим при организации вычислительного эксперимента можно использовать эффективные численные методы» [4].

3. «Разрабатываются алгоритм и программа решения задачи на ЭВМ. Программирование решений определяется теперь не только искусством и опытом исполнителя, а перерастает в самостоятельную науку со своими принципиальными подходами» [4].

4. «Проведение расчетов на ЭВМ. Результат получается в виде некоторой цифровой информации, которую далее необходимо будет расшифровать. Точность информации определяется при вычислительном эксперименте достоверностью модели, положенной в основу эксперимента, правильностью алгоритмов и программ (проводятся предварительные «тестовые» испытания)» [4].

5. Обработка результатов расчетов, их анализ и выводы. «На этом этапе могут возникнуть необходимость уточнения ММ (усложнения или, наоборот, упрощения), предложения по созданию упрощенных инженерных способов решения и формул, дающих возможности получить необходимую информацию более простым способом» [4].

В представленном исследовании эксперименты будут проводиться с использованием разработанного приложения, описанного в разделе 2.

В приложение будут вводиться исходные данные в виде: уравнений системы, границ интервалов и начальных условий. Далее приложение будет выводить графики решения, с помощью которых будет проводиться сравнение.

Так как в реализованных алгоритмах решения систем дифференциальных уравнений численными методами использовались системы линейных ОДУ из двух уравнений, основным методом исследования

будет, сравнение графиков решения одной систем двумя методами. Это позволит более наглядно продемонстрировать разницу между методами. «Для интегрирования дифференциальных уравнений в MATLAB предусмотрены функции: ode45, ode23, ode113, ode15s, ode23s, ode23t, ode23tb. Функции ode23, ode45 и ode113 предназначены для решения нежестких дифференциальных уравнений и систем. Все остальные функции предназначены для численного интегрирования жестких дифференциальных уравнений и систем».

За эталон будет браться решения системы ОДУ с использованием встроенного в MATLAB решателя ode113 — многошаговый метод Адамса-Башворта-Мултона переменного порядка.

Это адаптивный метод, который может обеспечить высокую точность решения».

Ниже будут представлены результаты работы приложения в виде графиков решения систем ОДУ.

Решение следующей системы:

$$\begin{cases} x' = -2y_1 + 5y_2 - x \\ y' = 3y_1 - y_2 \end{cases} \quad (27)$$

При использовании алгоритма Рунге-Кутты 4-го порядка решение данной системы будет состоять из следующего графика:

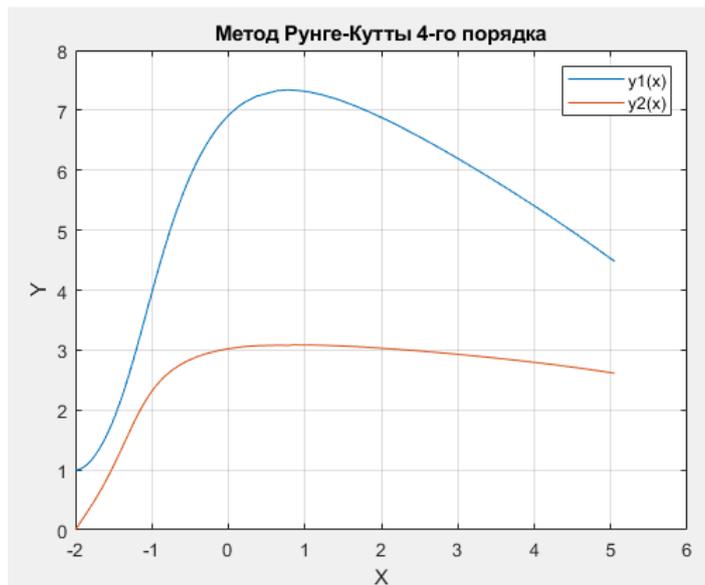


Рисунок 9 – График решения методом Рунге-Кутты 4-го порядка

При использовании алгоритма метода Эйлера решение данной системы будет состоять из следующего графика:

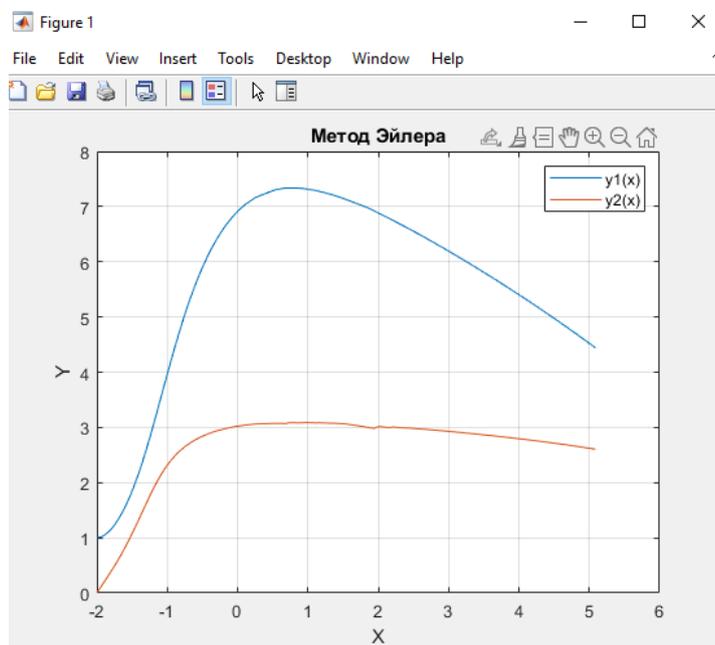


Рисунок 10 – График решения методом Эйлера

Решение следующей системы:

$$\begin{cases} x' = y_1 - 2y_2 - 7x \\ y' = 2y_1 + y_2 \end{cases} \quad (28)$$

При использовании алгоритма Рунге-Кутты 4-го порядка решение данной системы будет состоять из следующего графика

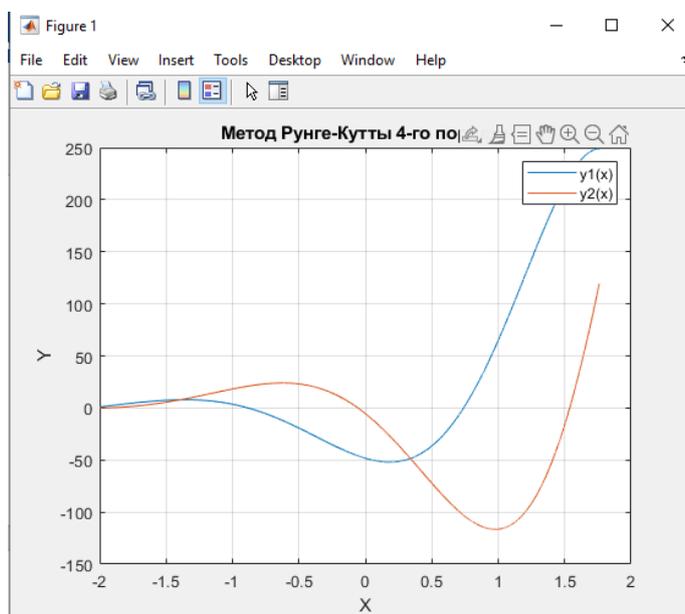


Рисунок 11 – График решения методом Рунге-Кутты 4-го порядка

При использовании алгоритма метода Эйлера решение данной системы будет состоять из следующего графика:

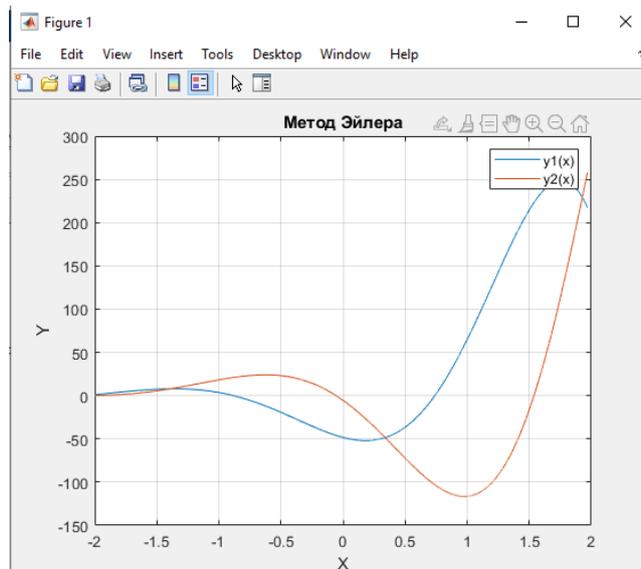


Рисунок 12 – График решения методом Эйлера

Решение следующей системы:

$$\begin{cases} x' = 6y_1 + 5y_2 - x \\ y' = y_1 - 9y_2 + x \end{cases} \quad (29)$$

При использовании алгоритма Рунге-Кутты 4-го порядка решение данной системы будет состоять из следующего графика:

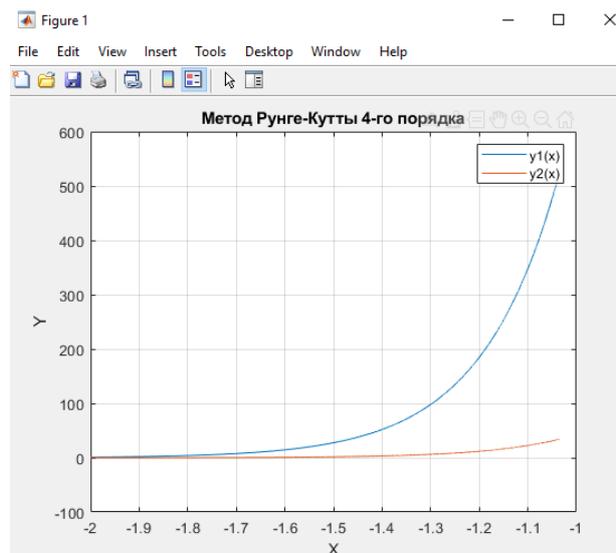


Рисунок 13 – График решения методом Рунге-Кутты 4-го порядка

При использовании алгоритма метода Эйлера решение данной системы будет состоять из следующего графика:

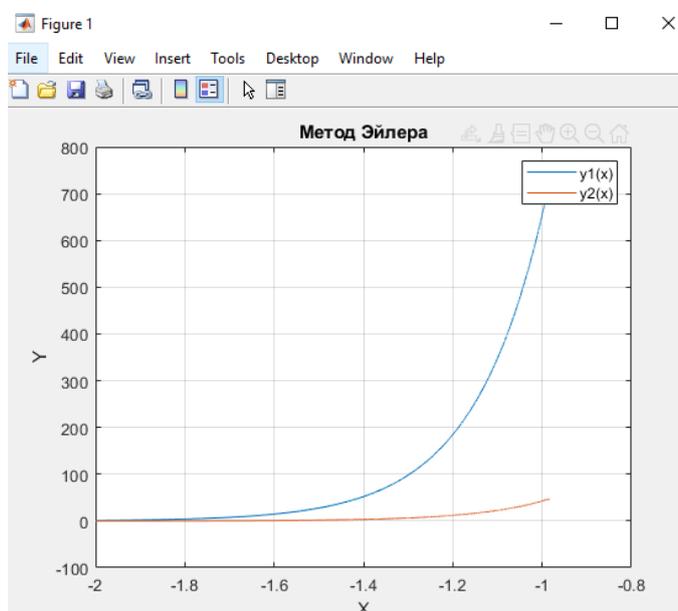


Рисунок 14 – График решения методом Эйлера

Посмотрев на полученные графики, нельзя не отметить, что метод Рунге-Кутты 4-го порядка имеет заметно большую точность решения системы ОДУ.

Воспользовавшись встроенным в Matlab решателем ode113, и сравнив с ним полученные результаты, можно понять, что решение методом Рунге-Кутты 4-го порядка, намного ближе к решению с помощью решателя. Тем временем как метод Эйлера куда больше менее похож на результат, полученный с помощью ode113. Благодаря этим данным можно сделать вывод, который будет описан в следующем подразделе. Исследование завершено.

3.2 Результаты эксперимента

Исходя из данных, полученных в ходе эксперимента, было выявлено что решение систем ОДУ самым распространенным методом Рунге-Кутты 4-го порядка, который является одним из наиболее применяемых, обеспечивает достаточно высокую точность в сравнении с методом Эйлера, ведь в отличие от метода Эйлера в методе Рунге-Кутта шаг делится не на 4 отрезка, в результате чего погрешность метода становится меньше.

График решения методом Рунге-Кутты 4-го порядка имеет большее сходство с графиком, построенным с помощью встроенного решателя ode113. Однако, в нашем случае, имея системы из двух линейных дифференциальных уравнений, разница в точности не столь велика для того, чтобы затрачивать больше усилий для реализации метода Рунге-Кутты 4-го порядка, поэтому метод Эйлера остается более предпочтительным за счет своей простоты в реализации.

3.3 Выводы и заключения по разделу 3

В этом разделе было протестировано приложение для решения систем дифференциальных уравнений численными методами, разработанное в предыдущем разделе этой бакалаврской работы.

Также с помощью этого приложения было решено несколько систем дифференциальных уравнений, приведены графики решения, проведен сравнительный анализ методов на основе результатов и на основе него сделаны выводы.

Заключение

В ходе данной работы был проведен анализ численных методов решения систем дифференциальных уравнений. В результате выяснялось, что наиболее удобные из предложенных методов, это метод Эйлера и Рунге-Кутты 4-го порядка. Таким образом было решено написать приложение, в котором будут реализованы эти два метода. Для написания данного приложения был выбран язык MATLAB.

Также были выполнены поставленные задачи:

- анализ численных методов;
- выбор методов для реализации приложения;
- выбор языка программирования;
- реализация приложения;
- тестирование разработанного приложения;
- проведение экспериментов и анализ полученных результатов.

Благодаря разработанному приложению было проведено исследование. Результатом исследования стало заключение о том, что метод Эйлера имеет более простую реализацию при меньшей точности, а метод Рунге-Кутты 4-го порядка большую точность при более сложной реализации.

Список используемых источников

1. Ведение в программные системы и их разработку [Электронный ресурс] : [учеб. пособие] / С. В. Назаров [и др.]. – 2-е изд., испр. – Москва : ИНТУИТ, 2016. – 649с. : ил.
2. Дифференциальные уравнения [Текст] : [учеб. пособие] / Н. Р. Жарова, Л. Г. Кузнецова ; М-во образования Российской Федерации, ГОУ ВПО "Нижевартовский гос. гуманитарный ун-т". - Нижневартовск : Изд-во Нижневартовского гос. гуманитарного ун-та, 2009. - 135 с. : ил., табл.
3. Дьяконов В. П. MATLAB. Полный самоучитель. – М.: ДМК Пресс, 2012. – 768 с.: ил. ISBN 978-5-94074-652-2
4. Краткий курс лекций по дисциплине : Математическое моделирование, численные методы и комплексы программ. [Электронный ресурс] [учеб. пособие]: – URL: <https://kubsau.ru/upload/iblock/baa/baabc92fb4e55a27c6b4521d971ff710.pdf>, Режим доступа: свободный (дата обращения: 17.05.2021)
5. Обзор численных методов [Электронный ресурс]: - URL: http://statistica.ru/branches-maths/obzor-chislennykh-metodov/?sphrase_id=103849, Режим доступа: свободный (дата обращения: 20.04.2021)
6. Основы работы в математическом пакете MathCAD / Алексеев Е.Р. [Электронный ресурс] [учеб. пособие]: – URL: <https://studfile.net/preview/4032223/>, Режим доступа: свободный (дата обращения: 20.04.2021)
7. Основы численных методов : [учеб. пособие] / Миньков С.Л., Миньков Л.Л. – Томск: НТЛ, 2006. — 261 с. — ISBN 5-89503-270-2.
8. Основы численных методов : [учеб. пособие] : / Л.И. Турчак, П.В. Плотников - 2-е изд. , перераб. и доп. - Москва : ФИЗМАТЛИТ, 2002. - 304 с. - ISBN 5-9221-0153-6.

9. Программирование и численные методы / Д.П. Костомаров. - М.: Московский государственный университет имени М.В. Ломоносова (МГУ), 2016. - 680 с.

10. Программирование численных методов / Юлий Кетков, Александр Кетков, Михаил Шульц. - М.: БХВ-Петербург, 2004. - 672 с.

11. Разработка веб-приложения : [Электронный ресурс]: - URL: https://studwood.ru/571274/informatika/razrabotka_prilozheniya, Режим доступа: свободный (дата обращения: 20.04.2021)

12. Решение обыкновенных дифференциальных уравнений с использованием MATLAB: [учеб. пособие] : / Л.Ф. Шампайн, И. Гладвел, С. Томпсон - 1-е изд. СПб.: Лань, 2009, 304 с.

13. Решение систем дифференциальных уравнений при помощи неявной схемы Адамса 3-го порядка : [Электронный ресурс]: - URL: <http://www.refsru.com/referat-7478-6.html>, Режим доступа: свободный (дата обращения: 12.05.2021)

14. Сборник материалов III региональной научно-практической конференции «Колмогоровские чтения-2007» [Электронный ресурс] – URL: <https://www.elibrary.ru/item.asp?id=22713825> (дата обращения: 17.05.2021)

15. Структура и Интерпретация Компьютерных Программ [Текст] / Х. Абельсон, Д. Д. Сассман. – М. : Добросвет, 2018. – 608 с. – ISBN 978- 5-98227-708-4.

16. Чарльз Генри Эдвардс, Дэвид Э. Пенни. Дифференциальные уравнения и краевые задачи: моделирование и вычисление с помощью Mathematica, Maple и MATLAB. 3-е издание. Киев.: Диалектика-Вильямс, 2007. ISBN 978-5-8459-1166-7.

17. Численное решение математических моделей объектов заданных системами дифференциальных уравнений / [Электронный ресурс] – URL: <https://habr.com/ru/post/418139/>, Режим доступа: свободный (дата обращения: 17.04.2021)

18. Численные методы / Под ред. Лапчика М.П.. - М.: Academia, 2017. - 608с.
19. Численные методы решения краевых задач для обыкновенных дифференциальных уравнений [Электронный ресурс] : [учеб. пособие] / А.Ю. Крайнов, К.М. Моисеева – Томский государственный университет, 2016. – 49с.- ISBN 978-5-93629-560-7.
20. Численные методы решения обыкновенных дифференциальных уравнений : [Электронный ресурс]: - URL: <https://docplayer.ru/39629131-4-chislennye-metody-resheniya-obyknovennyh-differencialnyh-uravneniy.html>,
Режим доступа: свободный (дата обращения:12.05.2021)
21. Элементарное введение. М.: кудиц-образ: / Н.Н. Мартынов 2005,- 416стр. EAN: 9785957900481.
22. David Skla. Learning PHP: A Gentle Introduction to the Web's Most Popular Language / David Skla. - O'Reilly, 2016. – 416с.
23. Deitel, H. Java How to Program / H. Deitel, P. Deitel. – 9th edition, Prentice Hall, 2015.
24. Hudson O. Getting started with IntelliJ IDEA // O. Hudson, Birmingham: Packt Publishing, 2013.
25. Krochmalski J. IntelliJ IDEA Essentials // J. Krochmalski. – Birmingham: Packt Publishing, 2014.
26. Masoud Kalali, Developing RESTful Services with JAX-RS 2.0, WebSockets, and JSON / Masoud Kalali – United Kingdom, Birmingham,: Packt Publishing, LTD, 2013.
27. Nixon R. Learning PHP, MySQL, JavaScript, CSS & HTML5 - 3rd Edition, 2014.

Приложение А

Исходный код разработанного программного обеспечения

Исходный код разработанного приложения для решения систем дифференциальных уравнений численными методами доступен на CD диске, приложенном к выпускной квалификационной работе.