**МИНОБРНАУКИ РОССИИ**

**САНКТ-ПЕТЕРБУРГСКИЙ ГОСУДАРСТВЕННЫЙ**

**ЭЛЕКТРОТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ**

**«ЛЭТИ» ИМ. УЛЬЯНОВА (ЛЕНИНА)**

**Кафедра алгоритмической математики**

**КУРСОВАЯ РАБОТА**

**по дисциплине «Дифференциальные уравнения»**

**Тема: Соскальзывание цепочки**

|  |  |  |
| --- | --- | --- |
| Студенты гр. 8382 |  | Кобенко В.П.  Черницын П.А. |
| Преподаватель |  | Павлов Д.А. |

Санкт-Петербург

2021

**ЗАДАНИЕ**

**НА КУРСОВУЮ РАБОТУ**

|  |  |  |
| --- | --- | --- |
| Студент Кобенко В.П.  Студент Черницын П.А. | | |
| Группа 8382 | | |
| Тема работы: Соскальзывание цепочки | | |
| Исходные данные:  Соскальзывание цепочки | | |
| Содержание пояснительной записки:   «Содержание», «Введение», «2-ой Закон Ньютона», «Прямой метод Ньютона», «Обратный метод Ньютона», «Метод Рунге-Кутты-Фельберга 4-5-го порядка», «Метод Хойна», «Метод Адамса», «Графический интерфейс», «Заключение», «Список использованных источников». | | |
| Предполагаемый объем пояснительной записки:  Не менее 20 страниц. | | |
| Дата выдачи задания: 09.05.2021 | | |
| Дата сдачи курсовой работы: 14.06.2021 | | |
| Дата защиты курсовой работы: 14.06.2021 | | |
| Студенты |  | Кобенко В.П.  Черницын П.А. |
| Преподаватель |  | Павлов Д.А. |

**АННОТАЦИЯ**

В курсовой работе рассмотрена задача соскальзывание цепочки. Для этого использовался 2-ой Закон Ньютона, его дифференциальная формулировка. Для решения поставленной задачи было использовано несколько методов: «Прямой метод Ньютона», «Обратный метод Ньютона», «Метод Рунге-Кутты-Фельберга 4-5-го порядка», «Метод Хойна», «Метод Адамса». Результаты решения данного уравнения были представлены в виде графиков в графическом интерфейсе.

**SUMMARY**

In the course work, the problem of chain slip. For this, the 2nd Newton law, its differential formulation, was used. To solve the problem, several methods were used: "Forward Newton's method", "Backward Newton's method", "Runge-Kutta-Felberg method of the 4-5th order", "Heun's method", "Adams method". The results of solving this equation were presented in the form of graphs in the graphical interface.

**СОДЕРЖАНИЕ**

[**ЗАДАНИЕ** 2](#_Toc75886730)

[**НА КУРСОВУЮ РАБОТУ** 2](#_Toc75886731)

[**АННОТАЦИЯ** 3](#_Toc75886732)

[**Введение** 5](#_Toc75886733)

[**2-ой Закон Ньютона** 6](#_Toc75886734)

[**Прямой метод Эйлера** 8](#_Toc75886735)

[**Обратный метод Эйлера** 10](#_Toc75886736)

[**Метод Рунге-Кутты-Фельберга 4-5 порядка** 12](#_Toc75886737)

[**Метод Хойна** 13](#_Toc75886738)

[**Метод Адамса-Башфорта** 14](#_Toc75886739)

[**GUI** 15](#_Toc75886740)

[**Вывод** 18](#_Toc75886741)

[**Используемая литература** 19](#_Toc75886742)

**Введение**

Дифференциальное уравнение является одним из фундаментальных понятий математики, широко применяемое в различных областях современных наук. Оно также применимо в физических процессах, один из которых рассматривается в данной курсовой работе. Соскальзывание цепочки является этим процессом. Были использованы методы интегрирования дифференциальных уравнений динамических систем для решения 2-ого закона Ньютона, такие как: «Прямой метод Ньютона», «Обратный метод Ньютона», «Метод Рунге-Кутты-Фельберга 4-5-го порядка», «Метод Хойна», «Метод Адамса».

**2-ой Закон Ньютона**

2-ой Закон Ньютона устанавливает связь между силой **F**, действующей на тело массы **m**, и ускорением **a**, которое приобретает тело под действием этой силы. Дифференциальная формулировка выглядит так:

Где **F** – сила, **t** – время, **p** – импульс.

В предположении, что движение одномерное, второй закон Ньютона в этом случае записывается в виде дифференциального уравнения второго порядка:

Согласно второму закону Ньютона, дифференциальное уравнение движения цепочки имеет вид:

Отсюда:

Поделим обе части уравнения на m:

Получаем:

Отсюда следует, что ускорение будет выглядеть так:

Проинтегрируем по dt и получим функцию скорости, использующуюся в численных методах.

Где - коэффициент трения, L – длина цепочки, g – ускорение свободного падения.

Для того, чтобы получить изменение скорости на шаге h, используется следующая формула:

После получения скорости *v* и ускорения *a* в программе используется формула для равноускоренного движения:

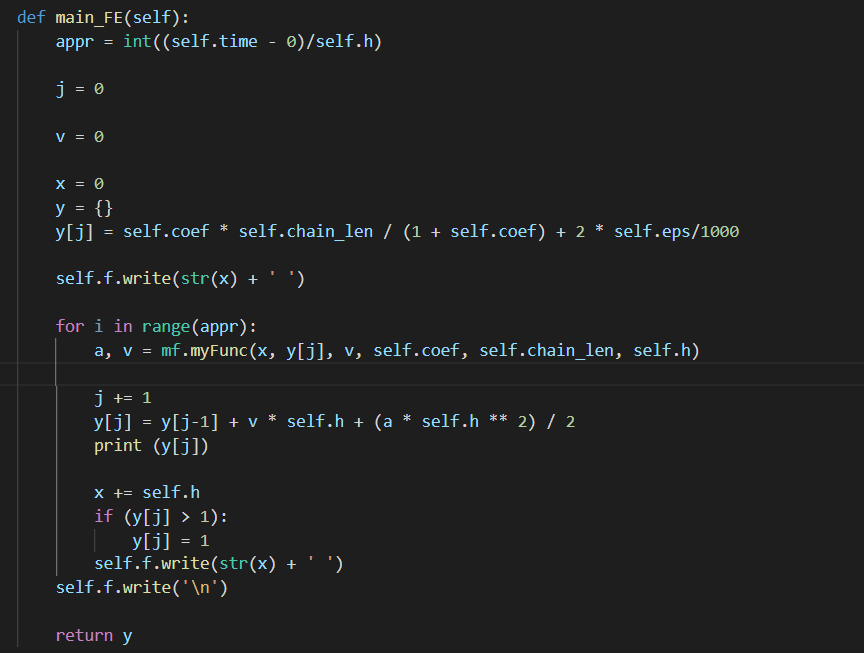
Для получения аналитической формулы должно быть известно начальное условие:

Длина свисающей части цепочки при равновесии составляет:

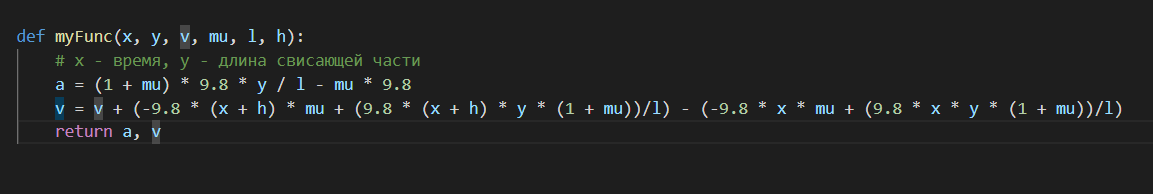
Скольжение цепочки описывается законом:

**Прямой метод Эйлера**

В нашем коде этот метод был реализован так:

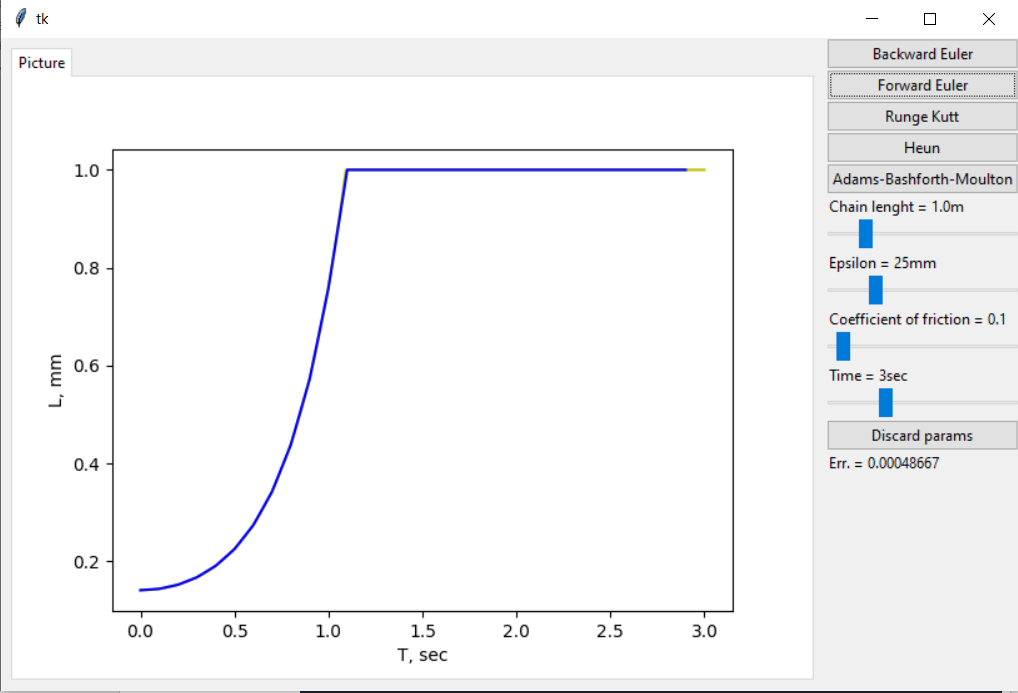


a,v – ускорение и скорость, полученные в результате выполнения функции f(x, t), self.h – это длина шага по x, my\_func.py выглядит так:

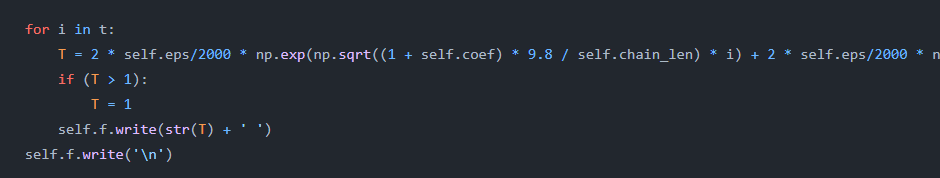


Где mu – это коэффициент трения, а 9.8 – ускорение свободного падения.

Для сравнения с аналитическим решением использовался график, желтая линия – график метода, синяя – аналитическое решение.



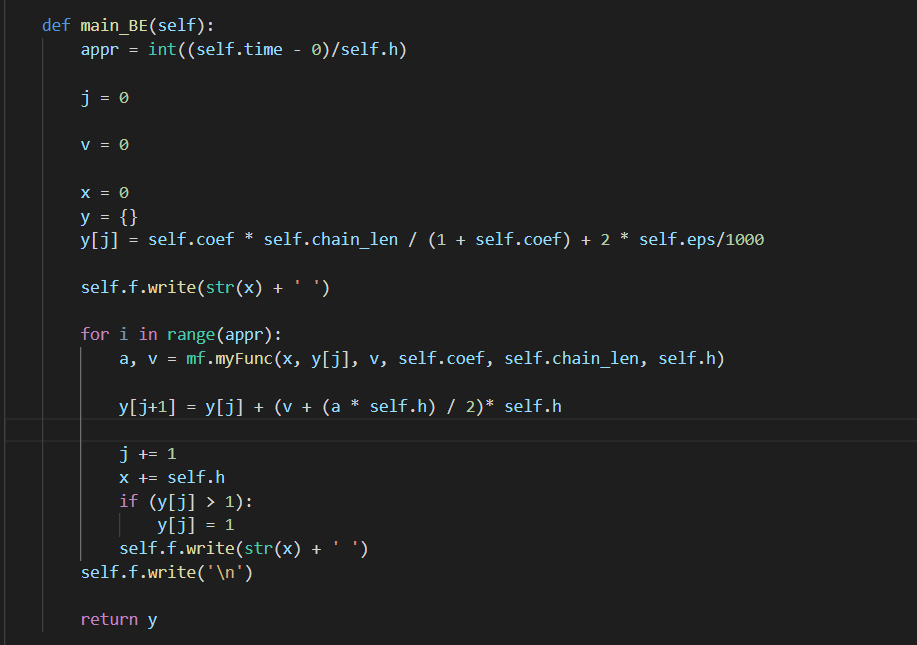
Реализация аналитического решения выглядит так (для всех методов она одинаковая):



Также после выполнения метода, на экран выводится его глобальная ошибка для шага h = 0.001.

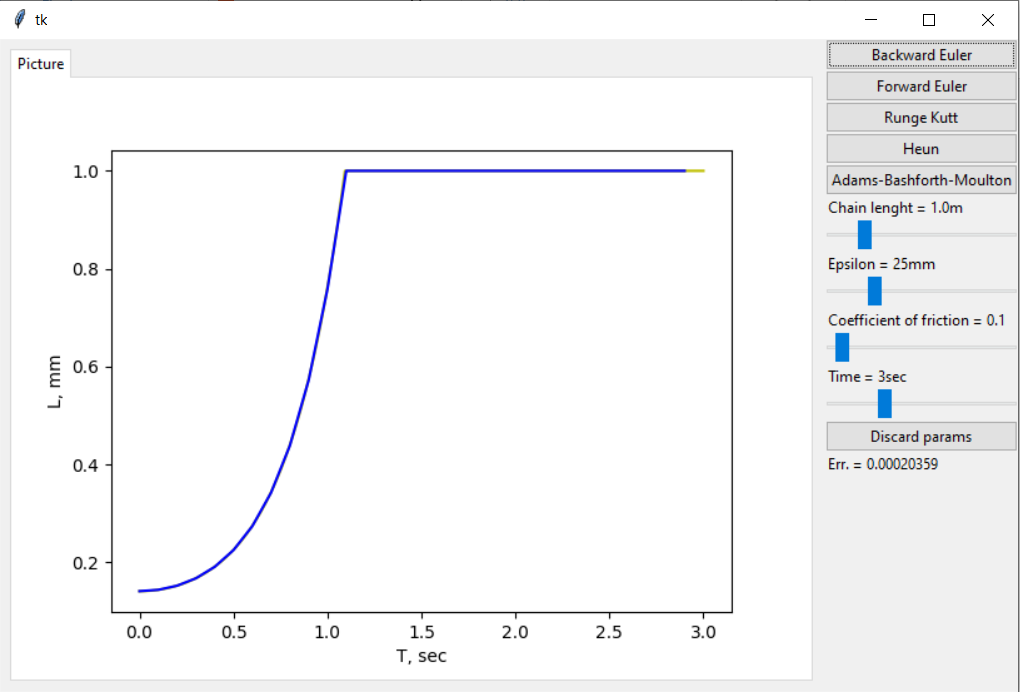
**Обратный метод Эйлера**

В нашем коде этот метод был реализован так:



Где a,v – ускорение и скорость, полученные в результате выполнения функции f(x, t), self.h – это длина шага по x.

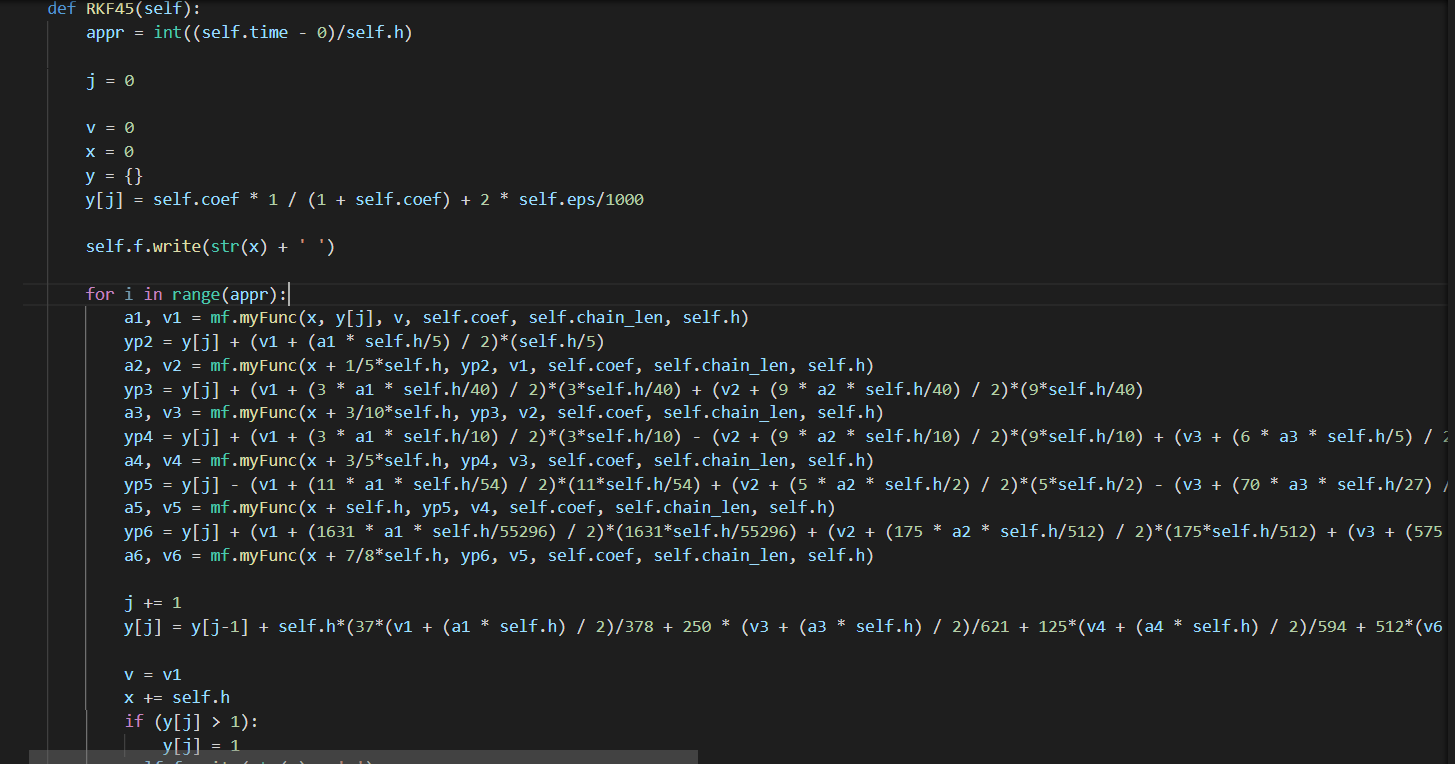
Для сравнения с аналитическим решением использовался график, желтая линия – график метода, синяя – аналитическое решение.



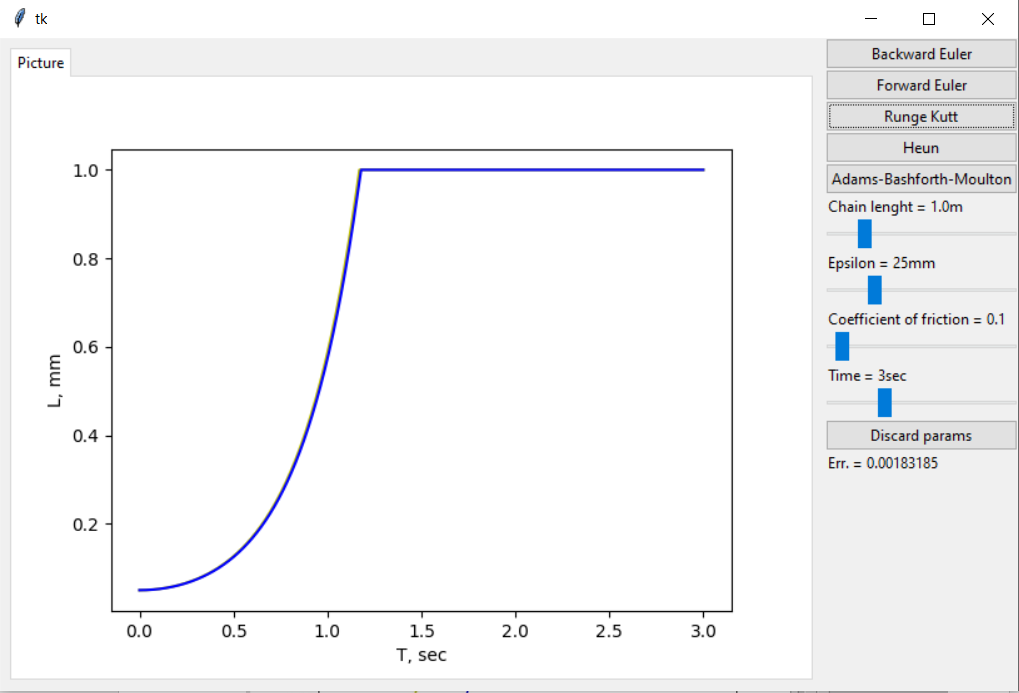
Также после выполнения метода, на экран выводится его глобальная ошибка для шага h = 0.001.

**Метод Рунге-Кутты-Фельберга 4-5 порядка**

Этот метод был реализован таким образом:



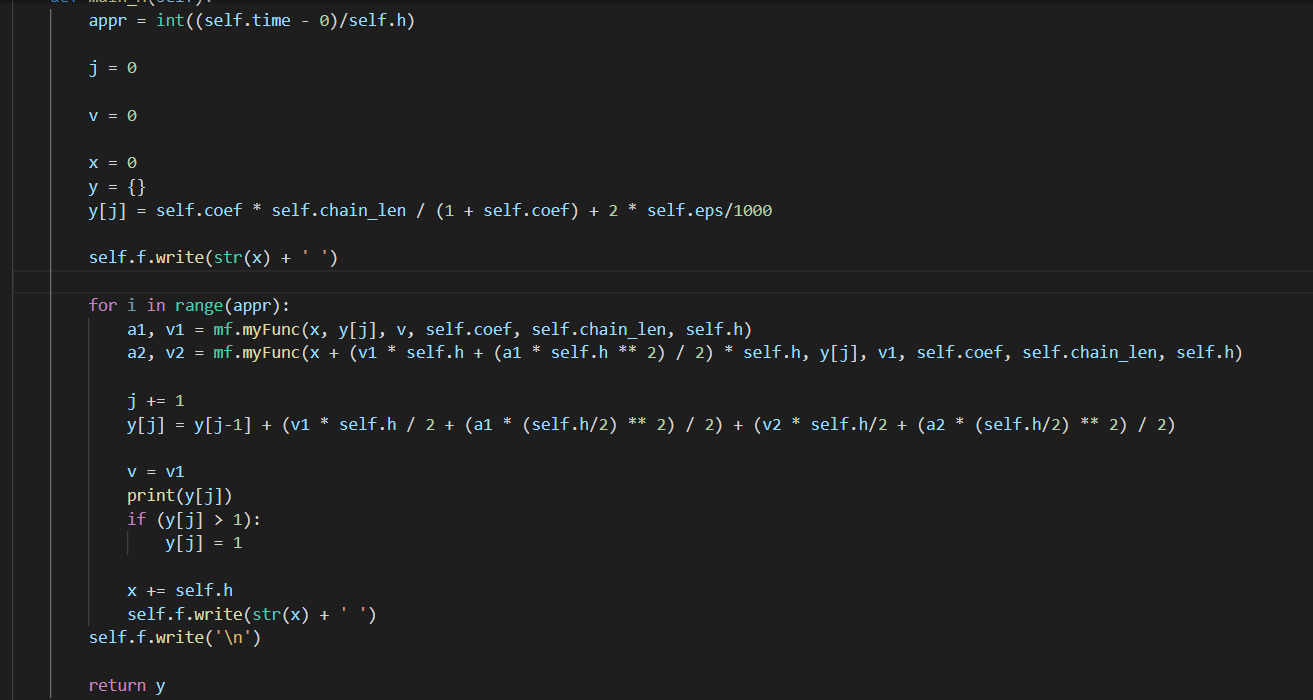
Для сравнения с аналитическим решением использовался график, желтая линия – график метода, синяя – аналитическое решение.



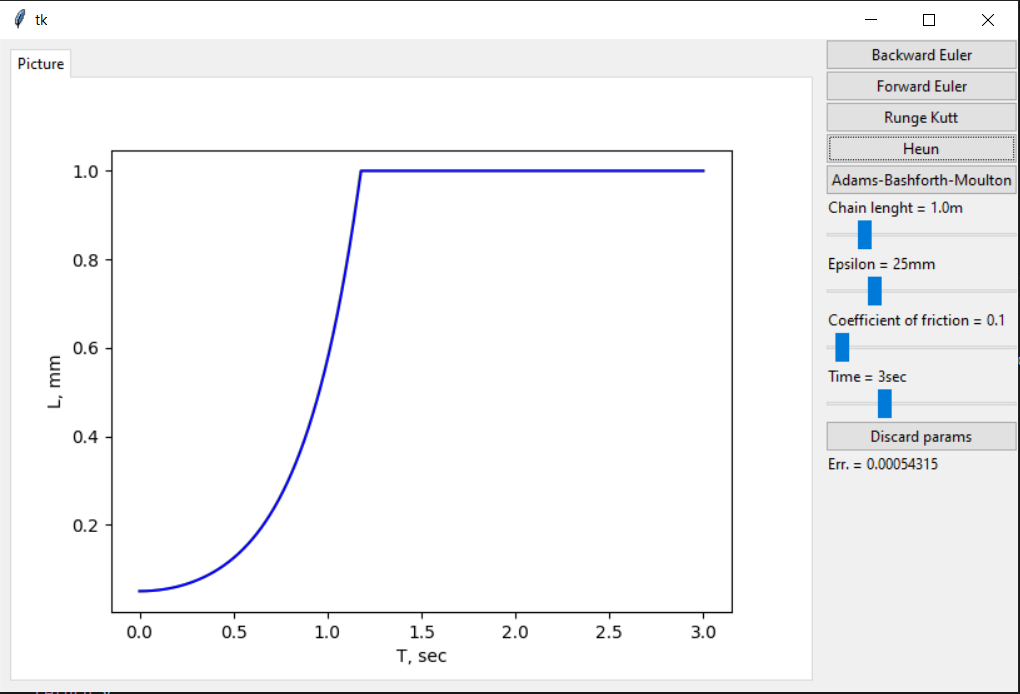
Также после выполнения метода, на экран выводится его глобальная ошибка для шага h = 0.001.

**Метод Хойна**

В работе метод был реализован так:



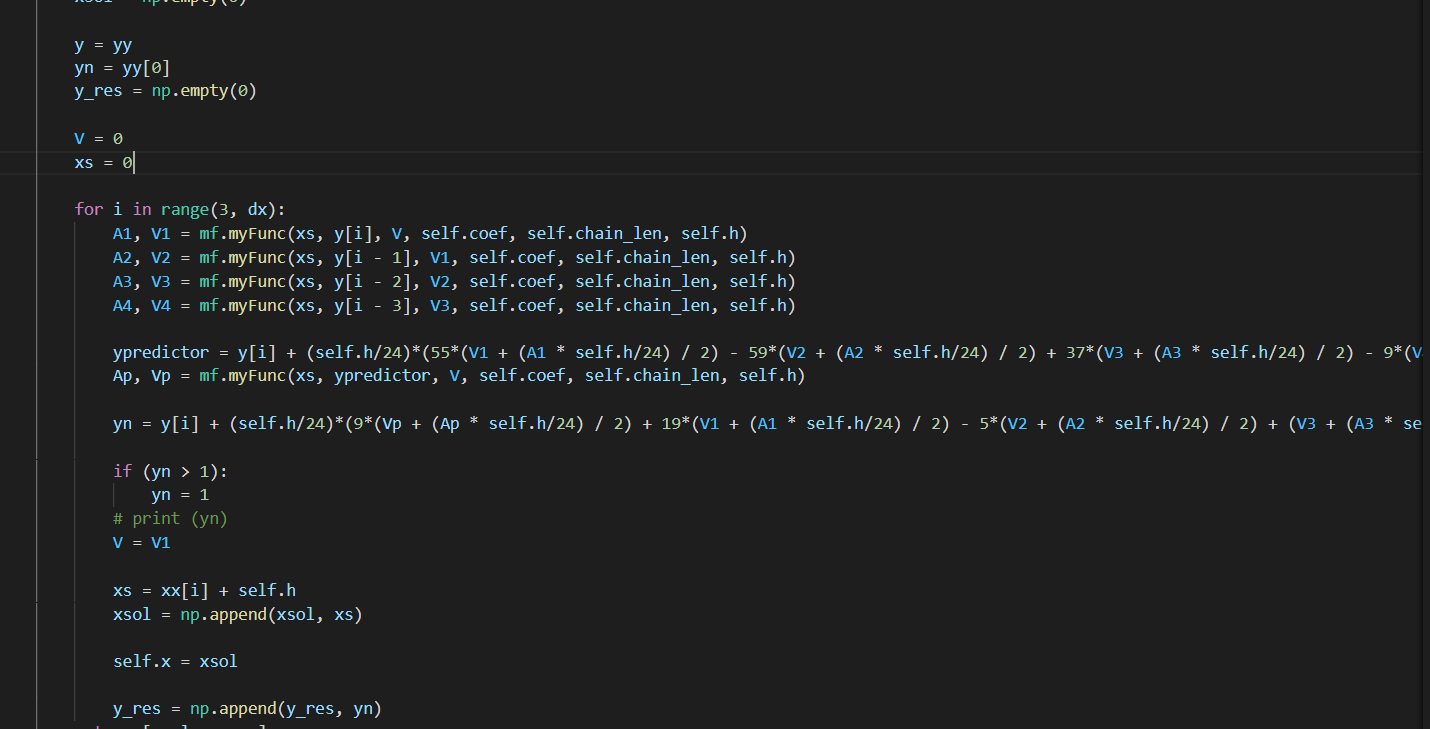
Для сравнения с аналитическим решением использовался график, желтая линия – график метода, синяя – аналитическое решение.



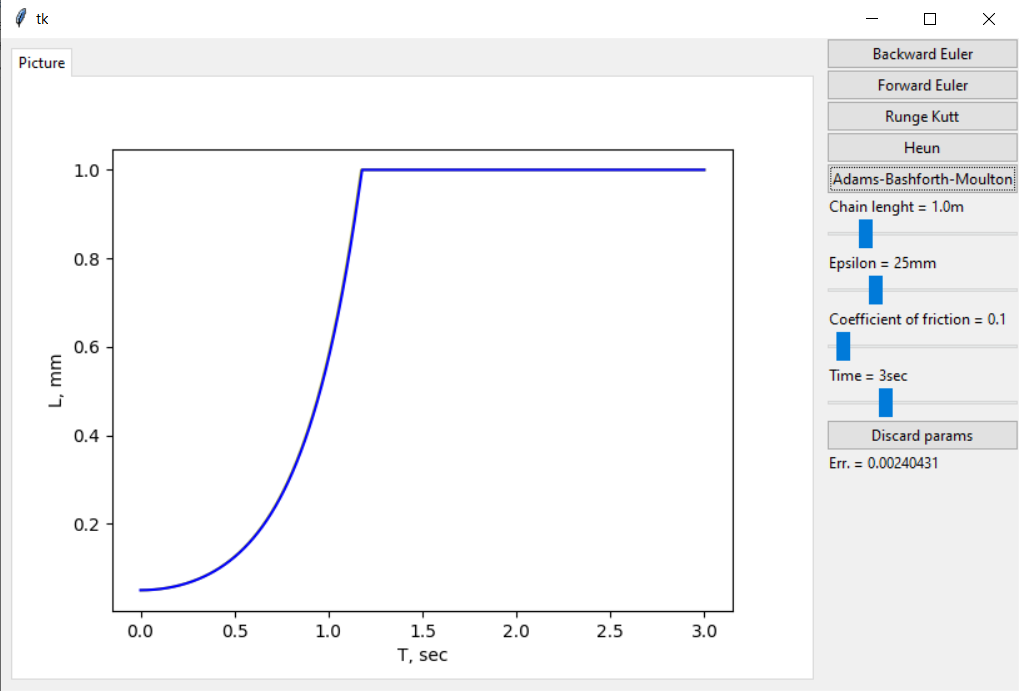
Также после выполнения метода, на экран выводится его глобальная ошибка для шага h = 0.001.

**Метод Адамса-Башфорта**

В работе метод был реализован так:



Для сравнения с аналитическим решением использовался график, желтая линия – график метода, синяя – аналитическое решение.



Также после выполнения метода, на экран выводится его глобальная ошибка для шага h = 0.001.

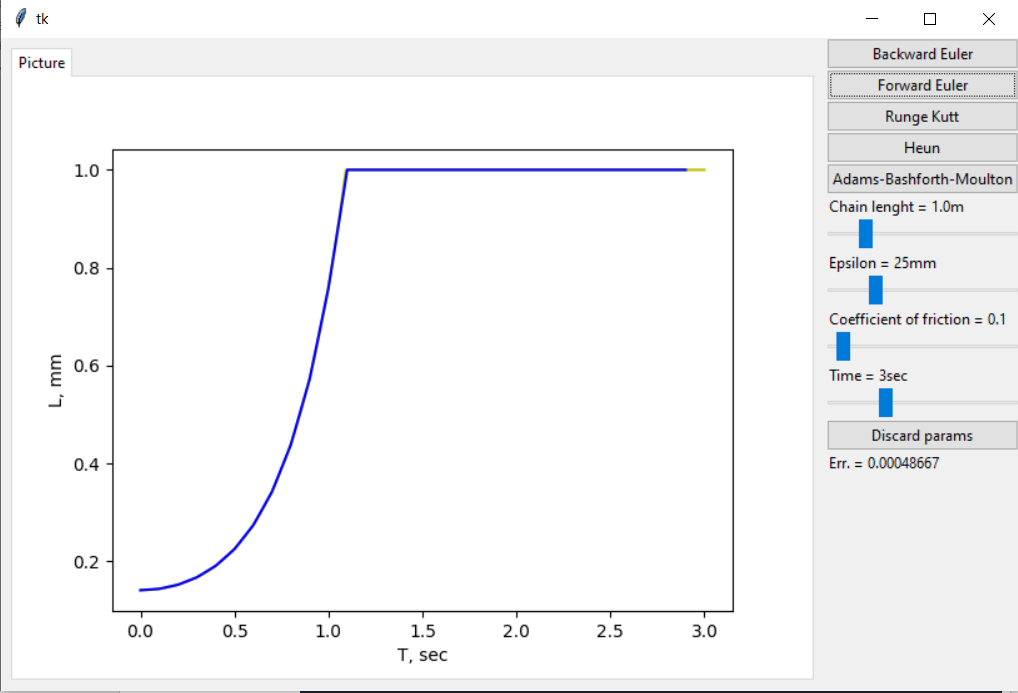
**GUI**

Был реализован графический интерфейс на языке Python с помощью библиотеки PySimpleGUI. Код программы представлен в приложении А.

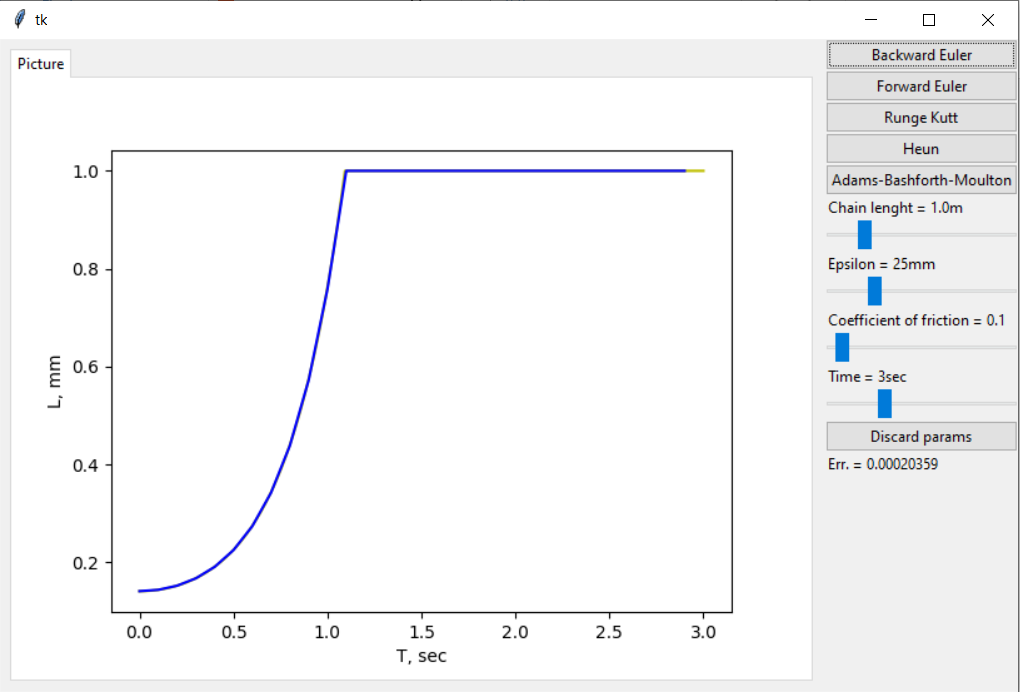
Интерфейс программы включает в себя 6 кнопок, 3 слайдера, окно для графиков и лэйбл с выводом информации об успешном/неуспешном запуске программы:

Первые пять кнопок в интерфейсе вызывают численные методы:

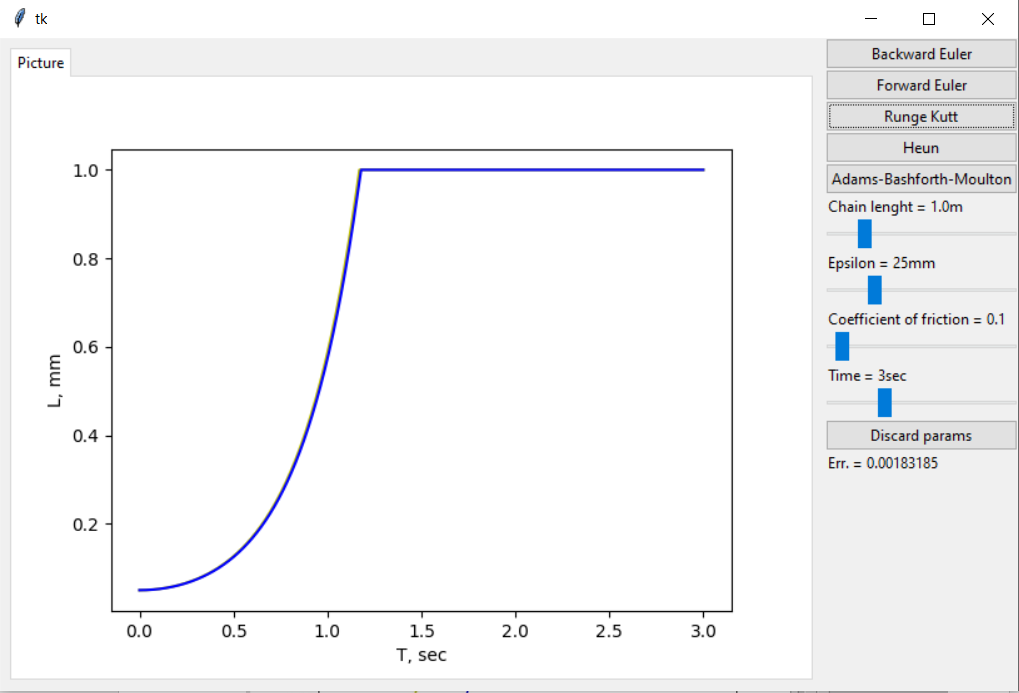
Forward Euler:



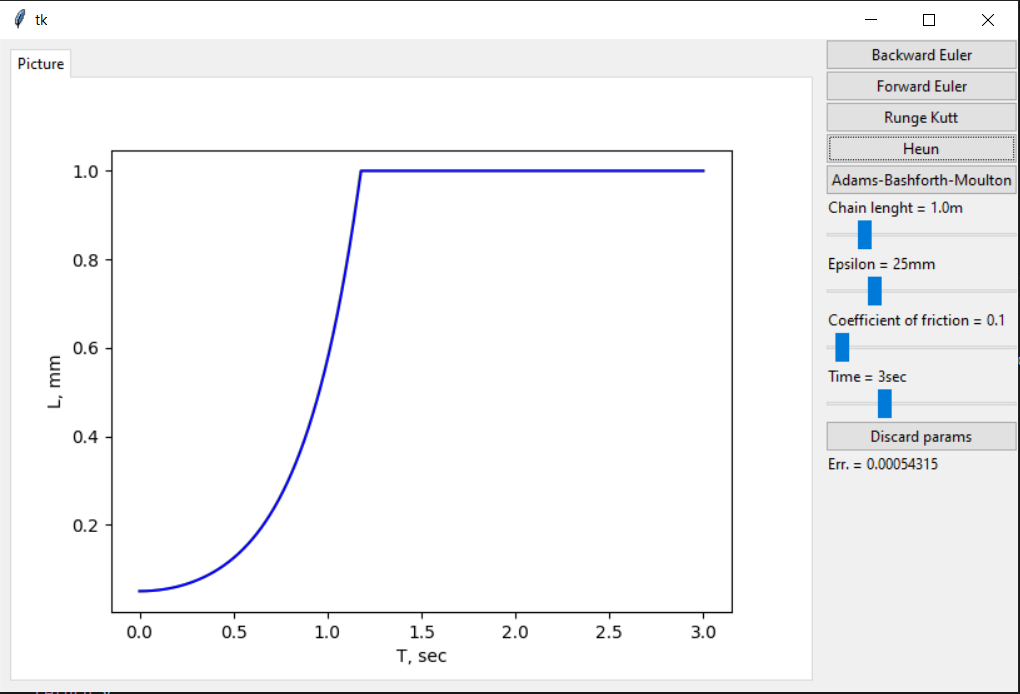
Backward Euler:



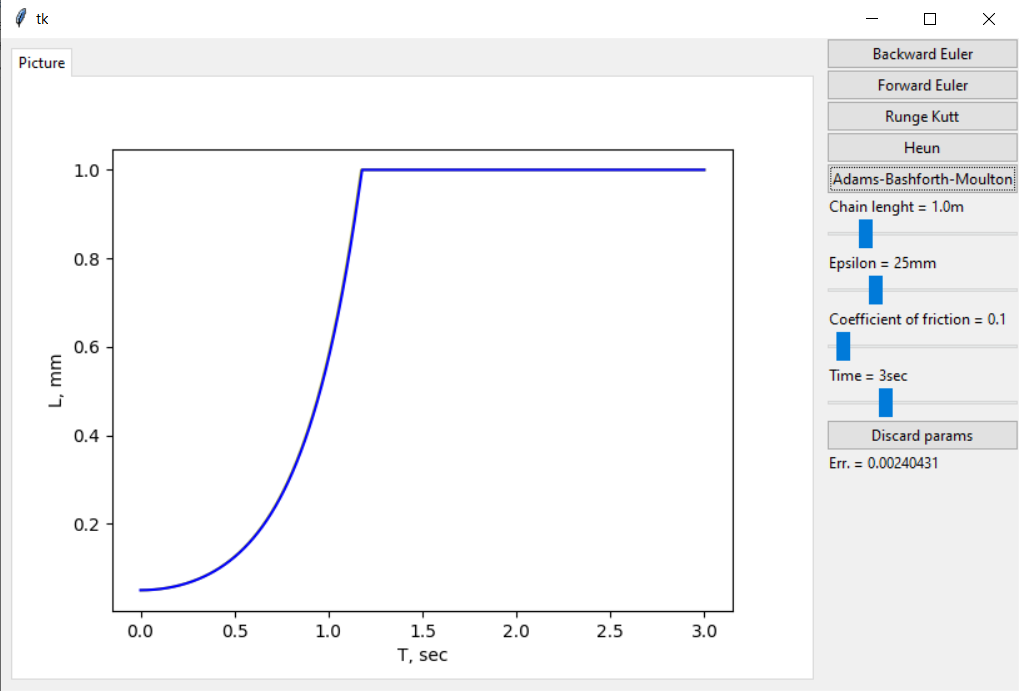
Runge Kutt:



Heun:



Adams-Bashfourth:



С помощью слайдеров можно менять условия поставленной задачи.

Chain length отвечает за длину цепочки, Epsilon за смещение цепочки относительно точки равновесия, Time – за время, кнопка Discard params – сбрасывает значения на исходные.

Также справа от графика выводится глобальная ошибка для конкретного метода с шагом h = 0.001

**Вывод**

В курсовой работе был рассмотрен процесс соскальзывания цепочки. Для решения задачи были использованы такие методы, как: «Прямой метод Ньютона», «Обратный метод Ньютона», «Метод Рунге-Кутты-Фельберга 4-5-го порядка», «Метод Хойна», «Метод Адамса». Было написано приложение на языке Python, решающее поставленную задачу данными методами и сравнивающее решение с аналитическим. Графический интерфейс позволяет увидеть отличия между методами на графиках.

**Используемая литература**

<https://old.math.tsu.ru/EEResources/pdf/diff_equation.pdf>

<http://math.smith.edu/~callahan/cic/ch4.pdf>

<https://en.wikipedia.org/wiki/Newton%27s_law_of_cooling>

<https://ru.wikipedia.org/wiki/%D0%9C%D0%B5%D1%82%D0%BE%D0%B4_%D0%90%D0%B4%D0%B0%D0%BC%D1%81%D0%B0>

<http://w.ict.nsc.ru/books/textbooks/akhmerov/nm-ode_unicode/1-3.html>

https://tftwiki.ru/wiki/Heun%27s\_method

<https://pysimplegui.readthedocs.io/en/latest/>

<https://www.python.org/>