Министерство науки и высшего образования Российской Федерации
федеральное государственное бюджетное образовательное учреждение
высшего профессионального образования
«Московский государственный технический университет имени
Н.Э. Баумана (национальный исследовательский университет)»
(МГТУ им. Н.Э. Баумана)

ФАКУЛЬТЕТ «Робототехники и комплексной автоматизации»
КАФЕДРА «Системы автоматизированного проектирования (РК-6)»

ОТЧЕТ О ВЫПОЛНЕНИИ ЛАБОРАТОРНОЙ РАБОТЫ

по дисциплине «Вычислительная математика»

Студент:	Антоненко Григорий Андреевич
Группа:	РК6-53Б
Тип задания:	Лабораторная работа №3
Тема:	Вынужденные колебания маятника (вариант №4)

Студент ____________ Антоненко Г. А.
подпись, дата Фамилия, И.О.

Преподаватель ____________ ____________
подпись, дата Фамилия, И.О.

Москва, 2023

Содержание

Вынужденные колебания маятника (вариант №4)

Задание

Дано ОДУ 2-го порядка:

$$\frac{d^2\theta}{dt^2} + 0.1\frac{d\theta}{dt} + \sin(\theta) = \cos(t), \tag{1}$$

где $\theta(t)$ обозначает угол отклонения маятника от вертикальной оси как функцию времени t.
Требуется (базовая часть):

1. Преобразовать данное ОДУ в систему ОДУ 1-го порядка

2. Разработать следующие функции, каждая из которых возвращает дискретную траекторию системы ОДУ с правой частью, заданной функцией f, начальным условием x_0, шагом по времени h и конечным временем t_n:
   - runge_kutta(x_0, t_n, f, h), где дискретная траектория строится с помощью явного метода Рунге-Кутта 4-го порядка;
   - adams_moulton(x_0, t_n, f, h), где дискретная траектория строится с помощью неявного трёхшагового метода Адамса-Моултона(выполняется в рамках продвинутой части);
   - milne_simpson(x_0, t_n, f, h), где дискретная траектория строится с помощью метода Милна-Симпсона (схема предиктор-корректор).

3. Для каждого из реализованных методов:
   - Численно каждым из методов найти траектории заданной динамической системы, используя шаг $h = 0.1$ и 15 различных начальных условий, для которых: $\theta(0) = 0$ и $\frac{d\theta}{dt}|_{t=0}$ следует выбрать случайно из интервала [1.85;2.1].
   - Вывести полученные траектории на едином графике как зависимости $\theta(t)$ (для каждого метода на отдельном графике).

4. В цем принципиальные отличия реализованных методов друг от друга? В чем они схожи?

5. Для каждой из схем каково значение шага, при котором она становится неустойчивой?

Цель выполнения лабораторной работы

Цель выполнения лабораторной работы - в рамках базовой части данной лабораторной работы требуется изучить численные методы решения задачи Коши для систем обыкновенных дифференциальных уравнений первого порядка, реализовать метод Рунге-Кутта четвертого порядка и метод Милна-Симпсона (схема предиктор-корректор и провести исследование траекторий, которые являются решениями задачи Коши на основе математической модели маятника.

Выполненные задачи

1. Было преобразовано заданное ОДУ в систему ОДУ 1-го порядка.

2. Были написаны следующие функции, каждая из которых возвращает дискретную траекторию системы ОДУ с правой частью, заданной функцией f, начальным условием x_0, шагом по времени h и конечным временем t_n:
   2.1. runge_kutta(x_0, t_n, f, h), где дискретная траектория строится с помощью метода Рунге – Кутта 4-го порядка;
   2.2. milne_simpson(x_0, t_n, f, h), где дискретная траектория строится с помощью метода Милна – Симпсона.

3. Для каждого из реализованных методов были численно найдены траектории заданной динамической системы, при этом был использован шаг h = 0.1 и 15 начальных условий с $\theta(0) = 0$ и $\frac{d\theta}{dt}$ выбранным случайно из интервала [1.85; 2.1].

4. Были выведены полученные траектории на едином графике как зависимость θ от времени.

5. Было проведено исследование в чем реализованные методи схожи и чем они различаются

6. Был найден шаг, при котором схема становится неустойчивой.

Базовая часть

1 Преобразование ОДУ 2-го порядка

Для преобразования данного ОДУ (1) в систему ОДУ первого порядка произвена замену первой производной $\frac{d\theta}{dt}$. Из полученных выражений записана следующую систему:

$$\begin{cases} \dfrac{d\theta}{dt} = y \\ \\ \dfrac{dy}{dt} = cos(t) - 0.1y - \sin(\theta) \end{cases} \tag{2}$$

2 Явный метод Рунге-Кутта 4-го порядка

Разработана функцию f, которая будет передаваться в качетсве одного из аргументов в функции runge_kutta(x_0, t_n, f, h) и milne_simpson(x_0, t_n, f, h)(листинг 1). Эта функция возвращает правую часть системы ОДУ 2.

Листинг 1. Функция f

```
def f(t, y):
  return np.array([y[1], -0.1*y[1] - np.sin(y[0]) + np.cos(t)])
```

В методах Рунге—Кутта в вычислениях участвуют значения приближенного решения только в двух соседних узлах w_i и w_{i+1}. Самый распространенный метод Рунге-Кутта - это метод четвертого порядка (RK4). Он включает в себя вычисление четырех коэффициентов на основе пробных точек в интервале и затем обновление значений функции в следующей точке с использованием этих коэффициентов. Метод Рунге-Кутта используют для расчета стандартных моделей достаточно часто, так как при небольшом объеме вычислений он обладает точностью метода $O(h^4)$. Метод Рунге-Кутта 4-го описывает следующим образом:

$$
\begin{gathered}
w_0 = \alpha, \\
k_1 = hf(t_i, w_i), \\
k_2 = hf\left(t_i + \frac{h}{2}, w_i + \frac{1}{2}k_1\right), \\
k_3 = hf\left(t_i + \frac{h}{2}, w_i + \frac{1}{2}k_2\right), \\
k_4 = hf(t_i + h, w_i + k_3), \\
w_{i+1} = w_i + \frac{1}{6}(k_1 + 2k_2 + 2k_3 + k_4), \quad i = 0, 1, \ldots, m-1,
\end{gathered}
$$

Для реализации вышеупомянутого метода, была разработана Python-функция runge_kutta(x_0, t_n, f, h) (листинг 2), реализующая метод Рунге-Кутта четвертого порядка, используя цикл с количеством итераций, соответствующим заданному шагу. На вход функции подаются переменные:

1. x_0 - начальное условие
2. t_n - конечное время
3. f - функция, описывающая дифференциальное уравнение
4. h - шаг по времени

Листинг 2. Функция runge_kutta(x_0, t_n, f, h)

```
def runge_kutta(x_0, t_n, f, h):
  steps = int(t_n / h)
  t = np.linspace(0, t_n, steps+1)
  w_r = np.zeros([steps+1, len(x_0)])
  w_r[0] = x_0

  for i in range(steps):
    k_1 = h*f(t[i], w_r[i])
    k_2 = h*f(t[i] + h/2, w_r[i] + k_1/2)
    k_3 = h*f(t[i] + h/2, w_r[i] + k_2/2)
    k_4 = h*f(t[i] + h, w_r[i] + k_3)
    w_r[i+1] = w_r[i] + (k_1 + 2*k_2 + 2*k_3 + k_4)/6

  return t, w_r
```

3 Метод Милна-Симпсона по схеме предиктор-корректор

Вторым реализованным методом являетяс метод Милна-Симпсона, который в отличие от метода Рунге-Кутта, требует не одно, а 4 начальных значения. Первым этапом в рассматриваемом методе является явный метод Милна, который имеет вид:

$$\omega_0 = \alpha_0, \quad \omega_1 = \alpha_1, \quad \omega_2 = \alpha_2, \quad \omega_3 = \alpha_3$$

$$\tilde{\omega}_{i+1} = \omega_{i-3} + \frac{4h}{3} \cdot (2 \cdot f(t_i, \omega_i) - f(t_{i-1}, \omega_{i-1}) + 2 \cdot f(t_{i-2}, \omega_{i-2})), \tag{3}$$

Втором этапом в рассматриваемом методе является неявный метод Симпсона, который имеет вид:

$$\omega_{i+1} = \omega_{i-1} + \frac{h}{3} \cdot (\tilde{f}(t_{i+1}, \tilde{\omega}_{i+1}) + 4 \cdot f(t_i, \omega_i) + f(t_{i-1}, \omega_{i-1})), \tag{4}$$

где $i = 3, \ldots, m-1$.
При комбинации вышеупомянтых методов(формулы 3 и 4) можно получить можно получить метод Милна-Симпсона(предиктор-корректор). Суммарная ошибка метода Милна-Симпсона есть величина порядка h^4, значит точность этого метода - $O(h^4)$ Для реализации вышеупомянутого метода, была разработана Python-функция milne_simpson(x_0, t_n, f, h) (листинг 3), реализующая метод Милна-Симпсона. На вход функции подаются переменные:

1. x_0 - начальное условие
2. t_n - конечное время
3. f - функция, описывающая дифференциальное уравнение
4. h - шаг по времени

Исходя из формулы 3 для данного метода необходимо 4 начальных значениях. Эти начальные значения вычисляются при помощи метода Рунге-Кутты, который был описан выше.

Листинг 3. Функция milne_simpson(x_0, t_n, f, h)

```python
def milne_simpson(x_0, t_n, f, h):
  steps = int(t_n / h)
  t = np.linspace(0, t_n, steps+1)
  w_m = np.zeros([steps+1, len(x_0)])
  w_m[0] = x_0
  t_start, w_start = runge_kutta(x_0, t_n, f, h)
  w_m[1] = w_start[1]
  w_m[2] = w_start[2]
  w_m[3] = w_start[3]

  for i in range(3, steps):
    w_cor = w_m[i-3] + 4 * h *(2*f(t[i] ,w_m[i]) - f(t[i-1], w_m[i-1]) + 2*f(t[i-2],
        w_m[i-2])) / 3
    w_m[i+1] = w_m[i-1] + h * (f(t[i+1], w_cor) + 4*f(t[i], w_m[i]) + f(t[i-1],
        w_m[i-1])) / 3

  return t, w_m
```

4 Вывод полученных траекторий как зависимостей $\theta(t)$

Функции для расчета траектории были описаны выше. Для их визуализации была использована библиотека matplotlib.pyplot из языка Python. Графики при значении конечного времени равном 150 можно наблюдать на рис. 1 и рис. 2. На графике разным цветом обозначены различные начальные условие, значения $\frac{d\theta}{dt}$ вынесены в легенду графика.

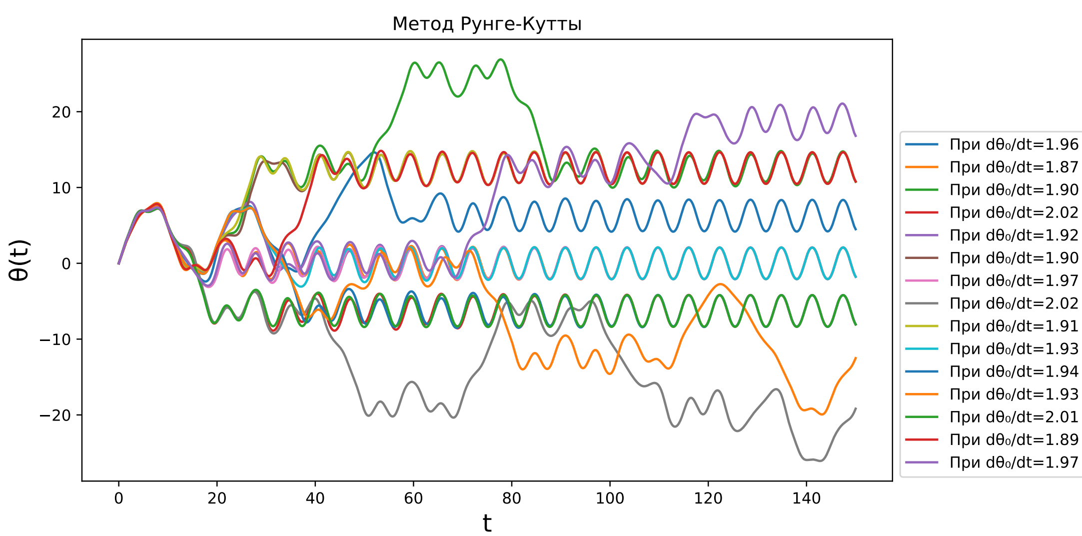

Рис. 1. Визуализация траекторий, рассчитанных методом Рунге-Кутта 4 порядка

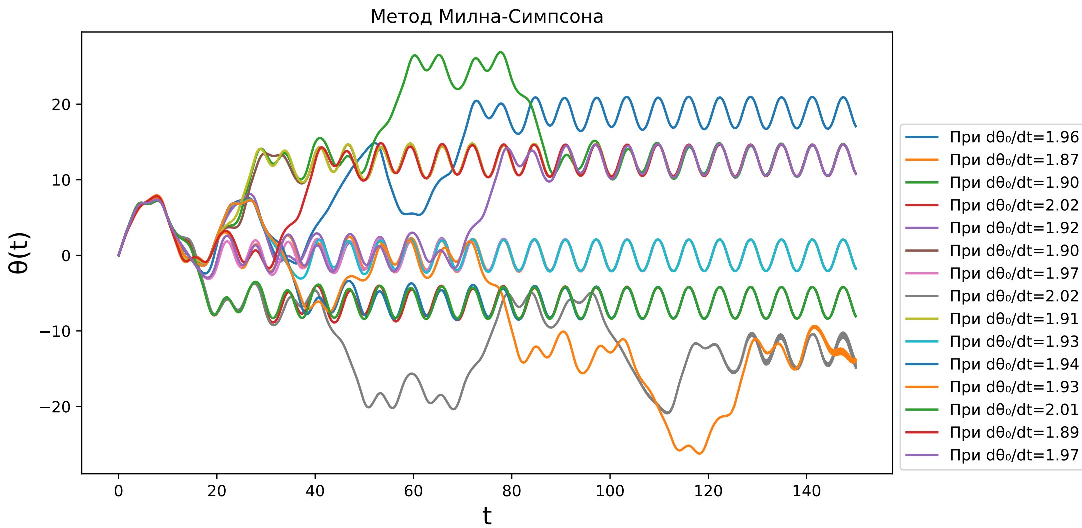

Рис. 2. Визуализация траекторий, рассчитанных методом Милна-Симпсона

5 Отличия и сходства методов

Произведено сравнение реализованных методов по нескольким критериям:

1. Количество шагов.
   Метод Рунге-Кутта 4-го порядка: одношаговый
   Метод Милна-Симпсона: многошаговый

2. Количетсво начальных значений необходимых для инициализации.
   Метод Рунге-Кутта 4-го порядка: одно значения
   Метод Милна-Симпсона: четыре значения

3. Классификация по явности/неявности метода. Метод Рунге-Кутта 4-го порядка: явный
   Метод Милна-Симпсона: явный и неявный

4. Точность.
   Метод Рунге-Кутта 4-го порядка: четвертый порядок точности
   Метод Милна-Симпсона: четвертый порядок точности

5. Применение.
   Метод Рунге-Кутта 4-го порядка: решение ОДУ и СОДУ
   Метод Милна-Симпсона: решение ОДУ и СОДУ

6 Определение шага, при котором каждая из схем становится неустойчивой

Численная схема становится неустойчивой, значит, что при увеличении значения шага (то есть при увеличении интервала между точками, на которых производятся численные вычисления), ошибка в результате вычислений начинает расти неограниченн. В других словах, схема перестает предоставлять корректные и надежные результаты. Тогда были исследованы траектории при разных шагах по времени, но одном и том же начальном условии.
Для нахождения шага, при котором метод Рунге-Кутта становится неусточивым рассматривались график траекторий при:
$h = [1.05, 0.98, 0.91, 0.84, 0.77, 0.7, 0.63, 0.56, 0.49, 0.42, 0.35, 0.28, 0.21, 0.14, 0.1]$.
График траекторий при различных значениях шага для метода Рунге-Кутта 4 порядка представлены на рис. 3 Исходя из сформированного графика, значения шага при котором схема неустойчива для метода Рунге-Кутта 4-го порядка: h = 1.05.

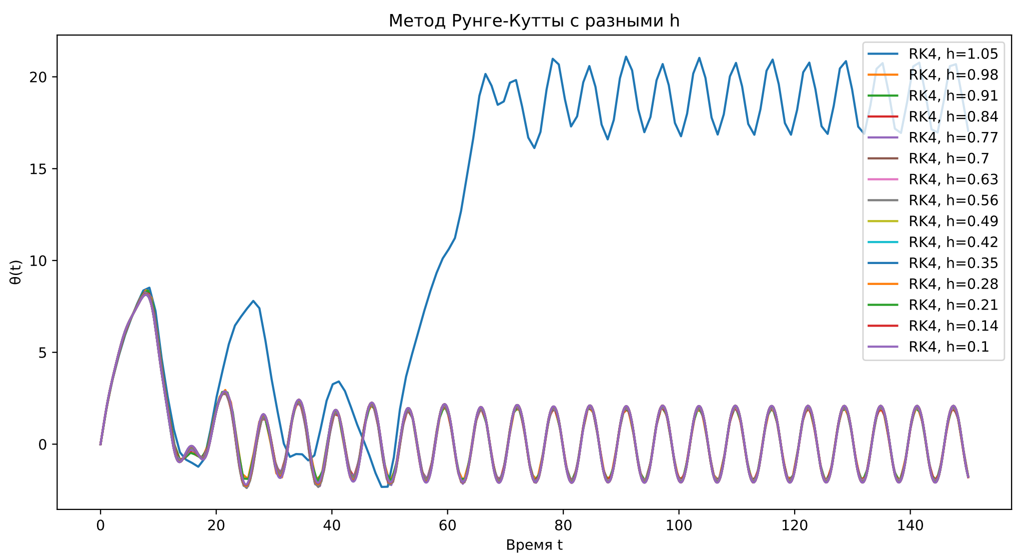

Рис. 3. График траекторий при различных значениях шага, полученный при помощи метода Рунге-Кутта

Для нахождения шага, при котором метод Милна-Симпслна становится неусточивым рассматривались график траекторий при:
$h = [0.49, 0.42, 0.35, 0.28, 0.21, 0.14, 0.1]$.
График траекторий при различных значениях шага для метода Милна-Симпсона представлены на рис. 4 Исходя из сформированного графика, значения шага при котором схема неустойчива начинается со значения h = 0.28.

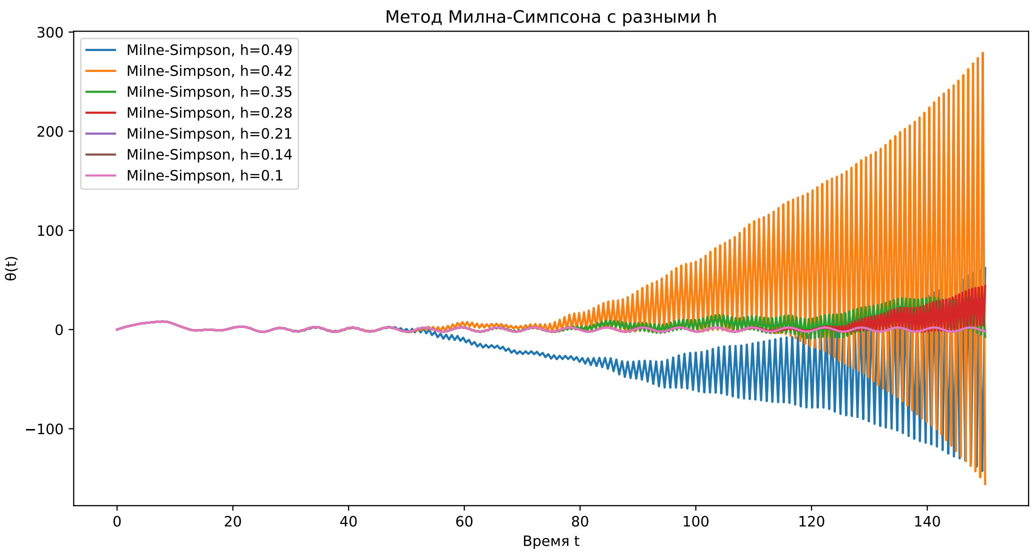

Рис. 4. График траекторий при различных значениях шага, полученный при помощи метода Милна-Симпсона

По итогу метод Рунге-Кутта 4-го порядка оказался устойчивее метода Милна-Симпсона. Зачастую, и правда, метод Рунге-Кутта 4-го порядка обычно более устойчив при больших значениях шага, что позволяет использовать более крупные шаги для численных вычислений. В то время как метод Милна-Симпсона (схема предиктор-корректор) требует меньших значений шага для обеспечения устойчивости, что может снижать эффективность.

Заключение

В процессе выполнения базовой части данной лабораторной работы, были проведены исследования методов решения задачи Коши для систем обыкновенных дифференциальных уравнений 1-го порядка. В частности, рассмотрены методы Рунге-Кутта 4-го порядка и метод Милна-Симпсона. Выполнен поиск и построение траекторий, отражающих изменение угла отклонения маятника от времени в случае вынужденных колебаний. Все это было достигнуто с применением численных методов для решения задачи Коши.

Список использованных источников

1. Першин А.Ю. Лекции по курсу «Вычислительная математика». Москва, 2018-2021. С. 140. URL: https://archrk6.bmstu.ru/index.php/f/810046.

2. Соколов, А.П. Инструкция по выполнению лабораторных работ (общая). Москва: Соколов, А.П., 2018-2021. С. 9. URL: https://archrk6.bmstu.ru. (облачный сер-

вис кафедры РК6).

3. Першин А.Ю., Соколов А.П., Гудым А.В. Сборник постановок задач на лабораторные работы по курсу «Вычислительная математика»: Учебное пособие. [Электронный ресурс]. Москва, 2023. С. 47. URL: hhttps://arch.rk6.bmstu.ru.

Выходные данные

Антоненко Г. А.Отчет о выполнении лабораторной работы по дисциплине «Вычислительная математика». [Электронный ресурс] — Москва: 2023. — 11 с. URL: https://sa2systems.ru:88 (система контроля версий кафедры РК6)

Постановка: *ассистент кафедры РК-6, PhD Першин А.Ю.*

Решение и вёрстка: *студент группы РК6-53Б, Антоненко Г. А.*

2023, осенний семестр