Федеральное государственное бюджетное образовательное учреждение высшего профессионального образования

«Московский государственный технический университет имени Н.Э. Баумана» (МГТУ им. Н.Э. Баумана)

Факультет «Информатика и системы управления»
Кафедра «Программное обеспечение ЭВМ и информационные технологии»

Отчет по лабораторной работе №2

по курсу:

«Моделирование»

Программно-алгоритмическая реализация методов Рунге-Кутта 2-го и 4-го порядков точности при решении системы ОДУ в задаче Коши.

Студент группы ИУ7-63Б: Н. О. Фурдик
(Фамилия И.О.)
Преподаватель: В. М. Градов
(Фамилия И.О.)

Москва, 2020

Оглавление

Задание

Цель работы: Получение навыков разработки алгоритмов решения задачи Коши при реализации моделей, построенных на системе ОДУ, с использованием методов Рунге-Кутта 2-го и 4-го порядков точности.

Исходные данные:

Задана система электротехнических уравнений, описывающих разрядный контур, включающий постоянное активное сопротивление, нелинейное сопротивление, зависящее от тока, индуктивность и емкость.

$$\begin{cases} \frac{dI}{dT} = \frac{U-(R_k+R_p(I))I}{L_k} \\ \frac{dU}{dt} = -\frac{I}{C_k} \end{cases} \quad (1)$$

Начальные условия:

$$t = 0, I = I_0, U = U_0, \quad (2)$$

где I, U - ток и напряжение на конденсаторе.

Сопротивление R_p можно рассчитать по формуле

$$R_p = \frac{l_p}{2\pi R^2 \int_0^1 \sigma(T(z))zdz} \quad (3)$$

Для функции $T(z)$ следует применить выражение

$$T(z) = T_0 + (T_w - T_0)z^m. \quad (4)$$

Параметры T_0, m находятся интерполяцией из табл. 1 при известном токе I.

Коэффициент электропроводности $\sigma(T)$ зависит от T и рассчитывается интерполяцией из табл.2.

Таблица 1

I, A	T_o, K	m
0.5	6730	0.5
1	6790	0.55
5	7150	1.7
10	7270	3
50	8010	11
200	9185	32
400	10010	40
800	11140	41
1200	12010	39

Таблица 2

T, K	σ, $\frac{1}{\text{Ом·см}}$
4000	0.031
5000	0.27
6000	2.05
7000	6.06
8000	12
9000	19.9
10000	29.6
11000	41.1
12000	54.1
13000	67.7
14000	81.5

Параметры разрядного контура:

$$R = 0.35, l_э = 12, L_k = 18710^{-6}, C_k = 26810^{-6} \quad (5)$$

$$R_k = 0.25, U_{co} = 1400, I_o = 0..3A, T_w = 2000K$$

Для справки: при указанных параметрах длительность импульса около 600 мкс, максимальный ток – около 800 А.

Задания к лабораторной

1. Построить графики зависимости отвремени импульса t: $I(t), U(t), R_p(t), I(t) \cdot R_p(t), T_0(t)$ при заданных выше параметрах. На одном из графиков привести результаты вычислений двумя методов разных порядков точности. Показать, как влияет выбор метода на шаг сетки.
2. График зависимости $I(t)$ при $R_k + R_p = 0$. Обратить внимание на то, что в этом случае колебания тока будут не затухающими.
3. График зависимости $I(t)$ при $R_k = 200$ Ом в интервале значений t $0 - 20$ мкс.

Листинг кода

В данном разделе будет представлен листинг кода программы.

Листинг 1: Логарифмическая интерполяция

```
def log_interp(x, y):
  log_x = np.log10(x)
  log_y = np.log10(y)
  lin_interp = interpolate.interp1d(logx, logy, fill_value='extrapolate')
  log_interp = lambda dd: np.power(10.0, lin_interp(np.log10(dd)))
  return log_interp
```

Листинг 2: Метод Рунге-Кутта 2-го порядка

```
def runge-kutta2():
  alpha=1.

  res_time = []
  res_curr = []
  res_volt = []
  res_res = []
  res_temp = []

  t = time_start()
  prev = initial_vector()

  res_time.append(t)
  res_cur.append(prev[0])
  res_volt.append(prev[1])
  res_res.append(lamp_resistance(prev[0]))
  res_temp.append(temperature(0, prev[0]))

  while t < time_end():
    v0 = vector_function(prev[0], prev[1])

    v1 = vector_function(prev[0] + step / (2.0 * alpha),
    prev[1] + step / (2.0 * alpha * v0[1]))
    prev[0] += step * ((1. - alpha) * v0[0] + alpha * v1[0])
    prev[1] += step * ((1. - alpha) * v0[1] + alpha * v1[1])

    res_time.append(t)
    res_cur.append(prev[0])
    res_volt.append(prev[1])
    res_res.append(lamp_resistance(prev[0]))
    res_temp.append(temperature(0, prev[0]))

    t += step

    print(t)

  return res_time, res_cur, res_volt, res_res, res_temp
```

Листинг 3: Метод Рунге-Кутта 4-го порядка

```
def runge-kutta4():
  res_time = []
  res_curr = []
  res_volt = []
  res_res = []
  res_temp = []

  t = time_start()
  prev = initial_vector()

  res_time.append(t)
  res_cur.append(prev[0])
  res_volt.append(prev[1])
  res_res.append(lamp_resistance(prev[0]))
  res_temp.append(temperature(0, prev[0]))

  while t < time_end():

  v1 = vector_function(prev[0], prev[1])
  k1 = v1[0]
  q1 = v1[1]

  v2 = vector_function(prev[0] + k1 * step / 2.0, prev[1] + q1 * step / 2.0)
  k2 = v2[0]
  q2 = v2[1]

  v3 = vector_function(prev[0] + k2 * step / 2.0, prev[1] + q2 * step / 2.0)
  k3 = v3[0]
  q3 = v3[1]

  v4 = vector_function(prev[0] + k3 * step, prev[1] + q3 * step)
  k4 = v4[0]
  q4 = v4[1]

  prev[0] += step * (k1 + 2 * k2 + 2 * k3 + k4) / 6.0
  prev[1] += step * (q1 + 2 * q2 + 2 * q3 + q4) / 6.0

  res_time.append(t)
  res_cur.append(prev[0])
  res_volt.append(prev[1])
  res_res.append(lamp_resistance(prev[0]))
  res_temp.append(temperature(0, prev[0]))

  t += step

  print(t)

  return res_time, res_cur, res_volt, res_res, res_temp
```

Результаты работы

Задание 1

Графики зависимости от времени импульса t при методе Рунге-Кутта 4-го порядка точности.

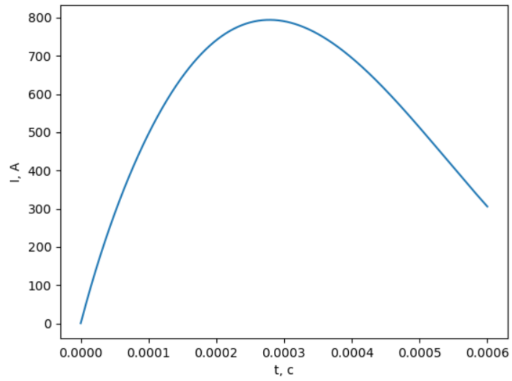

Рис. 1: График $I(t)$

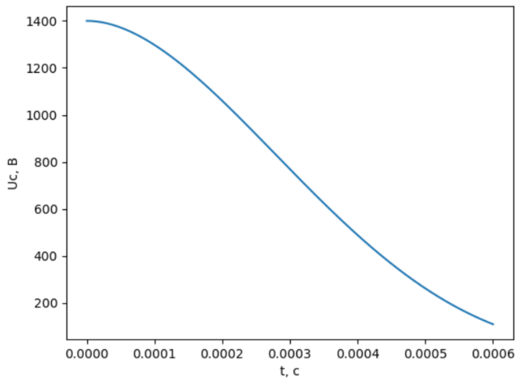

Рис. 2: График $U(t)$

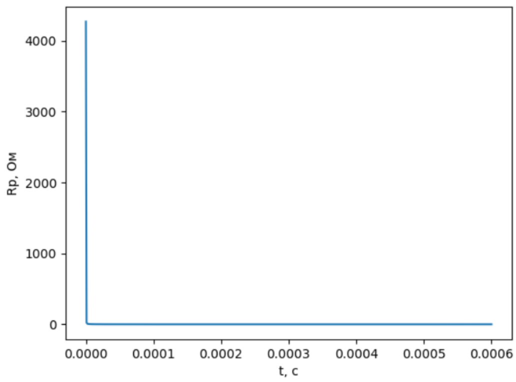

Рис. 3: График $R_p(t)$

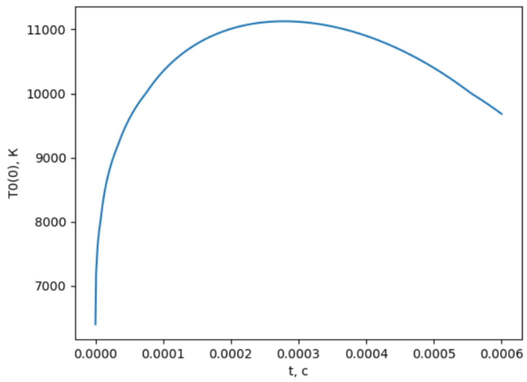

Рис. 4: График $I(t) \cdot R_p(t)$

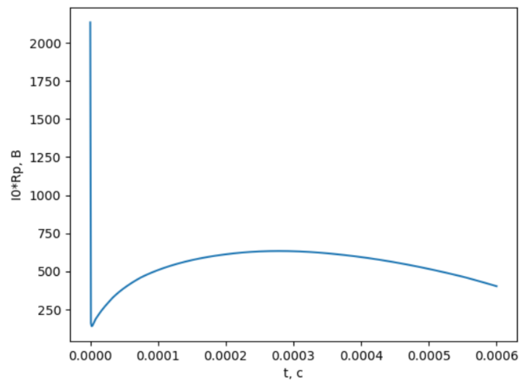

Рис. 5: График $T_0(t)$

Теперь построим для метода Рунге-Кнутта 2-го порядка точности график зависимости T_0 от t:

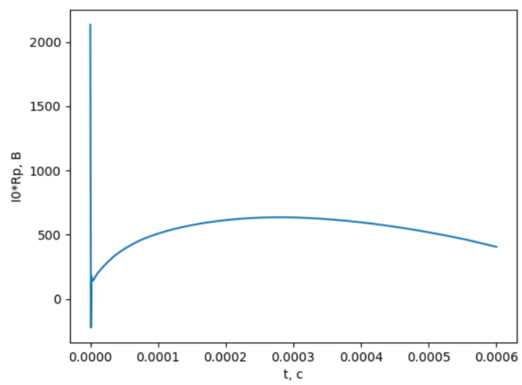

Рис. 6: График $T_0(t)$

Задание 2

График зависимости $I(t)$ при $R_k + R_p = 0$.

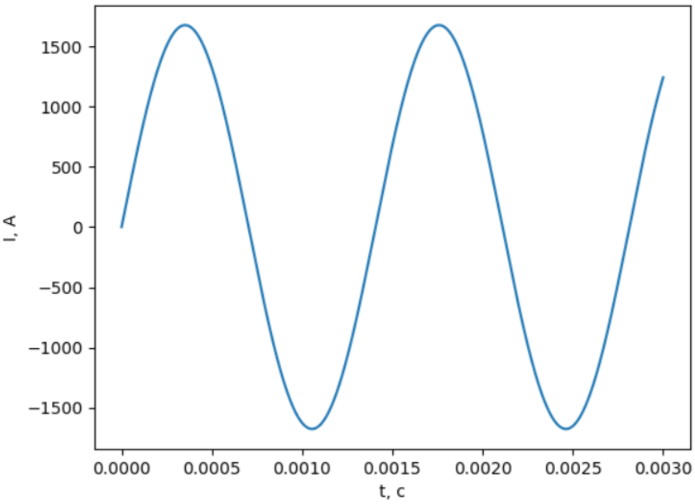

Рис. 7: График $I(t)$

Из графика видно, что колебания действительно стали незатухающими.

Задание 3

График зависимости $I(t)$ при $R_k = 200$ Ом в интервале значений t $0 - 20$ мкс.

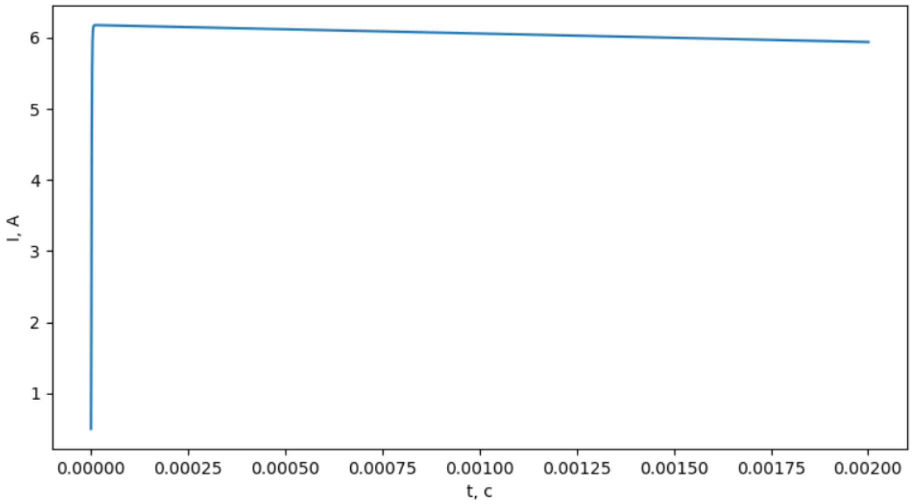

Рис. 8: График $I(t)$

Ответы на вопросы

1. Какие способы тестирования программы можно предложить?

Первый способ: поставить довольно большое сопротивление (200-250 Ом). В таком случае следует уменьшить шаг (например, до 1e-7), чтобы захватить момент резкого увеличения силы тока.

Второй способ: сделать R_k и R_p равными нулю. Тогда потерь в контуре не будет и на графике отобразится идеальная бесконечная синусоида.

Третий способ: сделать R_k равным какому-то небольшому константному значению (например, 1 Ом). В этом случае синусоида должна быть затухающей, а задачу следует решать аналитически.

2. Получите систему разностных уравнений для решения сформулированной задачи неявным методом трапеций. Опишите алгоритм реализации полученных уравнений.

Используя систему (1):

$$\begin{cases} \frac{dI}{dT} = \frac{U-(R_k+R_p(I))I}{L_k} \equiv f(I, U_c) \\ \frac{dU}{dt} = -\frac{I}{C_k} \equiv g(I) \end{cases} \tag{6}$$

Запишем выражения для метода:

$$\begin{cases} I_{n+1} = I_n + \Delta t \frac{f(I_n, U_{cn}) + f(I_{n+1}, U_{cn+1})}{2} \\ U_{cn+1} = U_{cn} + \Delta t \frac{g(I_n) + g(I_{n+1})}{2} \end{cases} \tag{7}$$

Подставим в (7) выражения f и g из (6):

$$\begin{cases} I_{n+1} = I_n + \Delta t \frac{U_{cn}-(R_k+R_p(I_n))I_n + U_{cn+1} - (R_k+R_p(I_{n+1}))I_{n+1})}{2L_k} \\ U_{cn+1} = U_{cn} - \Delta t \frac{I_n + I_{n+1}}{2C_k} \end{cases} \tag{8}$$

Подставим U_{cn+1} в верхнее уравнение и решим его относительно I_{n+1}:

$$I_{n+1} = \frac{-2C_k R_p(I_n) I_n \Delta t + 4C_k L_k I_n - 2C_k I_n R_k \Delta t + 4C_k U_{cn} \Delta t - I_n \Delta t^2}{4C_k L_k + 2C_k R_k \Delta t + 2C_k R_p(I_{n+1}) \Delta t + \Delta t^2} \tag{9}$$

Поскольку в правой части фигурирует $R_p(I_{n+1})$, следует решить это уравнение методом простой итерации (10).

$$x^{(s)} = f(x^{(s-1)}) \tag{10}$$

Полученное подставляем в (7) и находим U_{cn+1} по указанной формуле:

$$U_{cn+1} = U_{cn} + \Delta t \frac{g(I_n) + g(I_{n+1})}{2} \tag{11}$$

3. Из каких соображений проводится выбор того или иного метода, учитывая, что чем выше порядок точности метода, тем он более сложен?

Выбор зависит от степени точности метода, требуемой для конкретной задачи и от доступного количества ресурсов(объемов памяти)/времени.

Литература

1. Градов В.М. - Лекции по курсу "Моделирование"[Текст], Москва 2020 год.
2. Метод Рунге — Кутты [Электронный ресурс] - режим доступа: https://ru.wikipedia.org/wiki/Метод_Рунге_—_Кутты