



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

ФАКУЛЬТЕТ «Информатика и системы управления»

КАФЕДРА «Программное обеспечение ЭВМ и информационные технологии»

ОТЧЕТ ПО ЛАБОРАТОРНОЙ РАБОТЕ №2

Студент _____ Соколов Ефим _____
Группа _____ ИУ7-63Б _____
Дисциплина _____ Моделирование _____

Преподаватель:

_____ Градов В.М.
подпись, дата Фамилия, И.О.

Оценка _____

Москва — 2021 г.

Цель работы

Целью данной лабораторной работы является получение навыков разработки алгоритмов решения задачи Коши при реализации моделей, построенных на системе ОДУ, с использованием метода Рунге-Кутты 4-го порядка точности.

Постановка задачи

Задана система электротехнических уравнений, описывающих разрядный контур, включающий постоянное активное сопротивление R_k , нелинейное сопротивление $R_p(I)$, зависящее от тока I , индуктивность L_k и ёмкость C_k .

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

Начальные условия: $t = 0, I = I_0, U = U_0$ Здесь I, U - ток и напряжение на конденсаторе. Сопротивление R_p рассчитать по формуле:

$$R_p = \frac{l_k}{2\pi R^2 \int_0^1 \sigma(T(z))z dz}$$

Для функции $T(z)$ применить выражение $T_z = T_0 + (T_w - T_0)z^m$. Параметры T_0, m находятся интерполяцией из таблицы 1 при известном токе I . Коэффициент электропроводности $\sigma(T)$ зависит от T и рассчитывается интерполяцией из таблицы 2.

Таблица 1.

I, A	T ₀ , K	m
0.5	6730	0.50
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 , 1/Ом см
4000	0.031
5000	0.27
6000	2.05
7000	6.06
8000	12.0
9000	19.9
10000	29.6
11000	41.1
12000	54.1
13000	67.7
14000	81.5

Параметры разрядного контура:

$$R = 0.35 \text{ см}$$

$$l_e = 12 \text{ см}$$

$$L_k = 187 * 10^{-6} \text{ Гн}$$

$$C_k = 268 * 10^{-6} \text{ Ф}$$

$$R_k = 0.25 \text{ Ом}$$

$$U_{co} = 1400 \text{ В}$$

$$I_o = 0..3 \text{ А}$$

$$T_w = 2000 \text{ К}$$

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

Метод Рунге-Кутта 4-ого порядка

Порядок точности метода: $O(h^4)$.

$$y_{n+1} = y_n + \frac{k_1 + 2k_2 + 2k_3 + k_4}{6},$$

где

$$\begin{aligned} k_1 &= h_n \phi(x_n, y_n), \\ k_2 &= h_n \phi\left(x_n + \frac{h_n}{2}, y_n + \frac{k_1}{2}\right), \\ k_3 &= h_n \phi\left(x_n + \frac{h_n}{2}, y_n + \frac{k_2}{2}\right), \\ k_4 &= h_n \phi(x_n + h_n, y_n + k_3). \end{aligned}$$

Погрешность приведенной схемы Рунге-Кутта определяется максимальными значениями соответствующих производных. Оценку погрешностей можно получить для частного случая вида правой части дифференциального уравнения:

$$\phi(x, \mu) \equiv \phi(x).$$

Листинги кода

На листинге 1 приведена реализация метода Рунге-Кутта четвертого порядка на языке Python 3.

```
1 def runge_kutta_iv(l, U, R, h, Le, Lk, Rk, Ck):  
2     k1 = f(l, U, R, Le, Lk, Rk)  
3     m1 = g(l, Ck)  
4  
5     k2 = f(l + h * k1/2, U + h * m1/2, R, Le, Lk, Rk)  
6     m2 = g(l + h * k1/2, Ck)  
7  
8     k3 = f(l + h * k2/2, U + h * m2/2, R, Le, Lk, Rk)
```

```

9      m3 = g(l + h * k2/2, Ck)
10
11      k4 = f(l + h * k3, U + h * m3, R, Le, Lk, Rk)
12      m4 = g(l + h * k3, Ck)
13
14      ln = l + h * (k1 + 2*k2 + 2*k3 + k4) / 6
15      Un = U + h * (m1 + 2*m2 + 2*m3 + m4) / 6
16
17      return (ln, Un)

```

Листинг 1: Методы Рунге-Кутты четвертого порядка

На листинге 2 приведена реализация расчета $R_p(T)$ на языке Python 3.

```

1  def interpolate(table, table_y, y):
2      idx_max = 0
3      idx_min = 0
4      for i in range(len(table_y)):
5          if y > table_y[i]:
6              idx_max = i
7          else:
8              idx_max = i
9              break
10
11      if not idx_max:
12          idx_max = 1
13      idx_min = idx_max - 1
14
15      value = table[idx_min] + (table[idx_max] - table[idx_min]) /
16          (table_y[idx_max] - table_y[idx_min]) * (y - table_y[
17          idx_min])
18      return value
19
20  def f_l(l, z):
21      t0 = interpolate(table_T0, table_l, l)
22      m = interpolate(Table_M, table_l, l)
23
24      t = t0 + (tw - t0) * (z ** m)
25      sigma = interpolate(table_Sigma, table_T, t)
26
27      return sigma * z

```

```

27
28
29 def integ(l):
30     a = 0
31     b = 1
32     n = 100
33     h = (b - a) / n
34     s = (f_l(l, a) + f_l(l, b)) / 2
35     x = 0
36     for i in range(n - 1):
37         x += h
38         s += f_l(l, x)
39     s *= h
40
41     return s
42
43
44 def Rp(l, R, Le):
45     return Le / (2*pi * R*R * integ(l))

```

Листинг 2: Методы Рунге-Кутты четвертого порядка

Выполнение заданий лабораторной работы

Графики зависимости от времени импульса t : $I(t)$, $U(t)$, $R_p(t)$, произведения $I(t) * R_p(t)$, $T_0(t)$ при заданных выше параметрах. Указать шаг сетки

На рисунках 1-5 изображены графики зависимости $I(t)$, $U(t)$, $R_p(t)$, произведения $I(t) * R_p(t)$, $T_0(t)$.

Шаг сетки: $1/2 * e^{-6}$.

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

На рисунке 6 приведен график зависимости $I(t)R_k + R_p = 0$.

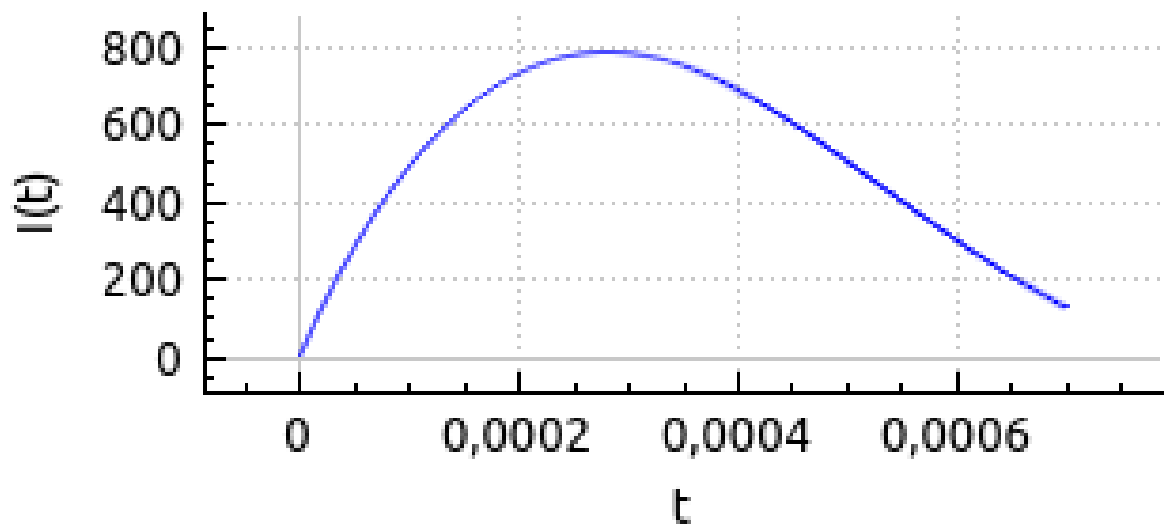


Рисунок 1: График зависимости $I(t)$

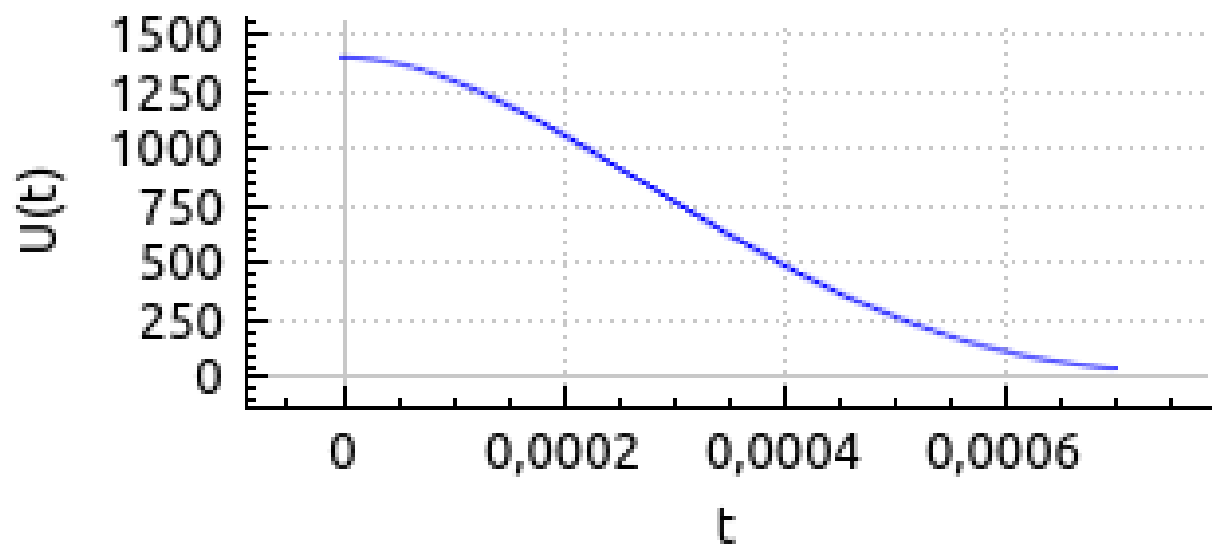


Рисунок 2: График зависимости $U(t)$

График зависимости $I(t)$ при $R_k + R_p = \text{const} = 200$ Ом в интервале значений t 0-20 мкс

На рисунке 7 приведен график зависимости $I(t)R_k + R_p = \text{const} = 200$ в интервале значений t 0-20 мкс.

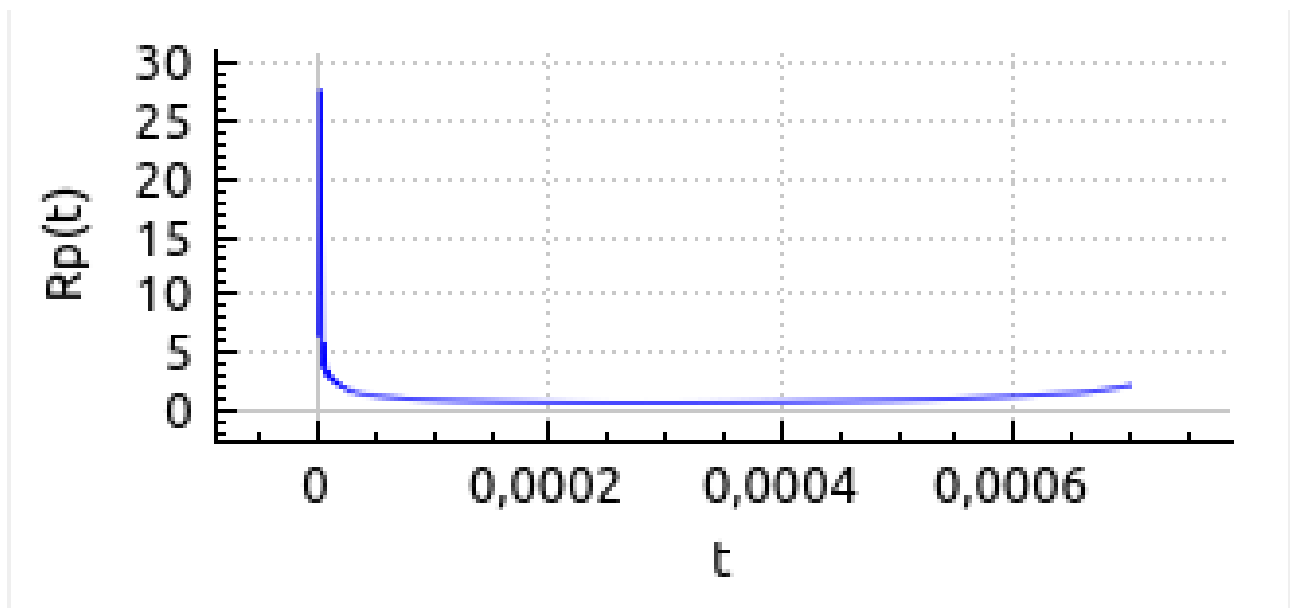


Рисунок 3: График зависимости $R_p(t)$

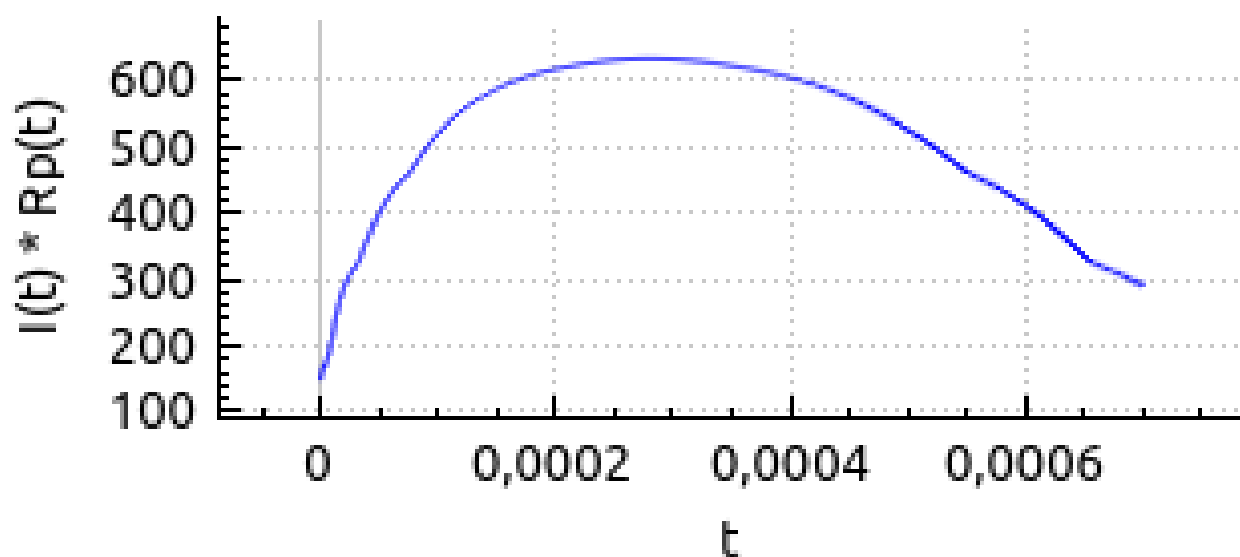


Рисунок 4: График зависимости произведения $I(t) * R_p(t)$

Результаты исследования влияния параметров контура C_k , L_k , R_k на длительность импульса t имп. аperiodической формы

Длительность импульса определяется по кривой зависимости тока от времени на высоте $0.35I_{max}$.

Изначальные замеры:

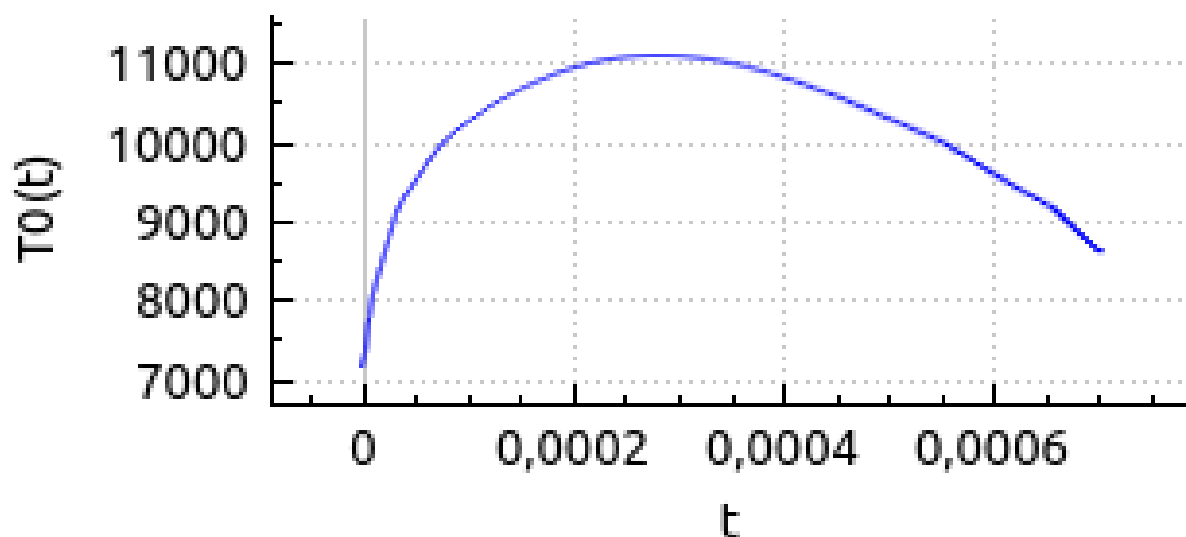


Рисунок 5: График зависимости $T_0(t)$

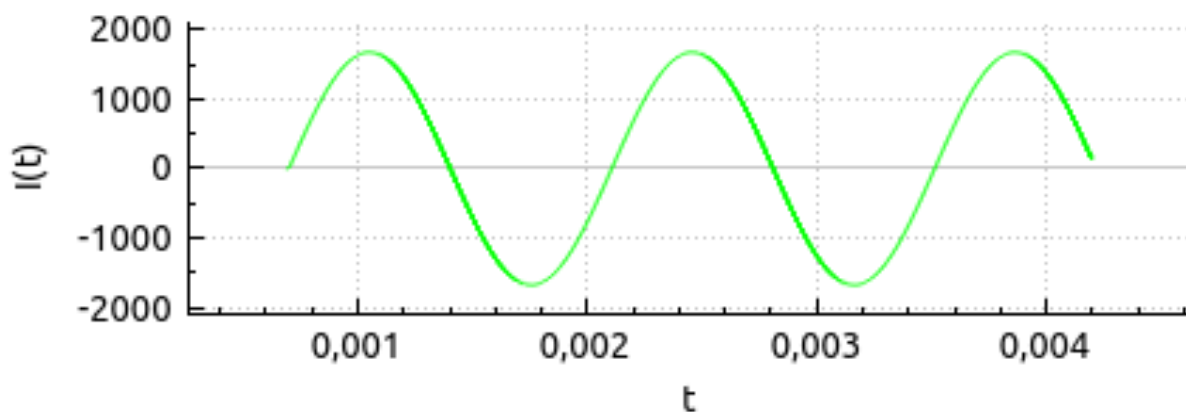


Рисунок 6: График зависимости $I(t)$ при $R_k + R_p = 0$

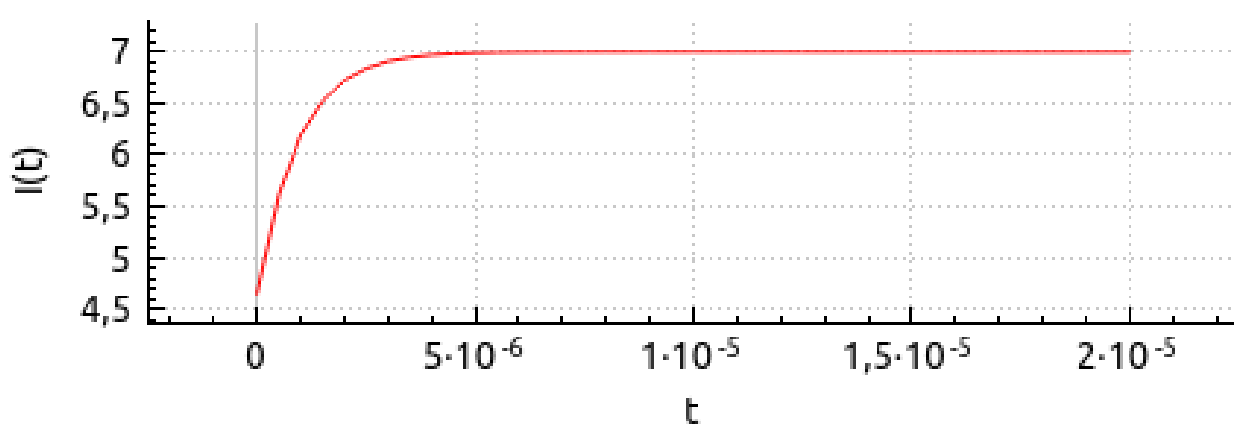


Рисунок 7: График зависимости $I(t)$ при $R_k + R_p = \text{const} = 200$

- $C_k = 268 * 10^{-6} \text{ Ф};$
- $L_k = 187 * 10^{-6} \text{ Гн};$
- $R_k = 0.25 \text{ Ом}.$

При заданных параметрах получим следующие величины:

- $I_{max} = 788.175 \text{ А}, 0.35I_{max} = 275.861 \text{ А};$
- $t_{imp_start} = 47,5 * 10^{-6} \text{ с}, t_{imp_stop} = 613 * 10^{-6} \text{ с};$
- $t_{imp} = t_{imp_stop} - t_{imp_start} = 565,5 * 10^{-6} \text{ с}.$

Исследование влияния параметра C_k

Уменьшим значение C_k : пусть $C_k = 150 * 10^{-6} \text{ Ф}$. При таких параметрах получим:

- $I_{max} = 654.55 \text{ А}, 0.35I_{max} = 229.092 \text{ А};$
- $t_{imp_start} = 38,5 * 10^{-6} \text{ с}, t_{imp_stop} = 456 * 10^{-6} \text{ с};$
- $t_{imp} = t_{imp_stop} - t_{imp_start} = 417,5 * 10^{-6} \text{ с}.$

Увеличим значение C_k : пусть $C_k = 300 * 10^{-6} \text{ Ф}$. При таких параметрах получим:

- $I_{max} = 838.551 \text{ А}, 0.35I_{max} = 293.493 \text{ А};$
- $t_{imp_start} = 5,1 * 10^{-6} \text{ с}, t_{imp_stop} = 683 * 10^{-6} \text{ с};$
- $t_{imp} = t_{imp_stop} - t_{imp_start} = 632 * 10^{-6} \text{ с}.$

На основании проделанных вычислений можно сделать вывод, что длительность импульса прямо пропорциональна C_k : при уменьшении C_k длительность импульса уменьшается и, наоборот, при увеличении C_k длительность импульса увеличивается.

Исследование влияния параметра L_k

Уменьшим значение L_k : пусть $L_k = 100 * 10^{-6}$ Гн. При таких параметрах получим:

- $I_{max} = 940.811$ А, $0.35I_{max} = 329,284$ А;
- $t_{imp_start} = 3,1 * 10^{-6}$ с, $t_{imp_stop} = 455,5 * 10^{-6}$ с;
- $t_{imp} = t_{imp_stop} - t_{imp_start} = 424,5 * 10^{-6}$ с.

Увеличим значение L_k : пусть $C_k = 230 * 10^{-6}$ Гн. При таких параметрах получим:

- $I_{max} = 738.55$ А, $0.35I_{max} = 258,493$ А;
- $t_{imp_start} = 5,4 * 10^{-6}$ с, $t_{imp_stop} = 678,5 * 10^{-6}$ с;
- $t_{imp} = t_{imp_stop} - t_{imp_start} = 624,5 * 10^{-6}$ с.

На основании проделанных вычислений можно сделать вывод, что длительность импульса прямо пропорциональна L_k (так же, как и C_k): при уменьшении L_k длительность импульса уменьшается и, наоборот, при увеличении L_k длительность импульса увеличивается.

Исследование влияния параметра R_k

Уменьшим значение R_k : пусть $R_k = 0,1$ Ом. При таких параметрах получим:

- $I_{max} = 870.66$ А, $0.35I_{max} = 304,731$ А;
- $t_{imp_start} = 5,2 * 10^{-6}$ с, $t_{imp_stop} = 607,5 * 10^{-6}$ с;
- $t_{imp} = t_{imp_stop} - t_{imp_start} = 555,5 * 10^{-6}$ с.

Увеличим значение R_k : пусть $R_k = 2,2$ Ом. При таких параметрах получим:

- $I_{max} = 596.373$ А, $0.35I_{max} = 208,731$ А;
- $t_{imp_start} = 3,6 * 10^{-6}$ с, $t_{imp_stop} = 667 * 10^{-6}$ с;

- $t_{imp} = t_{imp_stop} - t_{imp_start} = 631 * 10^{-6} \text{ с.}$

На основании проделанных вычислений можно сделать вывод, что длительность импульса прямо пропорциональна R_k (так же, как и C_k и L_k): при уменьшении R_k длительность импульса уменьшается и, наоборот, при увеличении R_k длительность импульса увеличивается.

Ответы на контрольные вопросы

1. Какие способы тестирования программы, кроме указанного в п.2, можете предложить ещё?

1. Провести тестирование при разном значении шага: если при очередном уменьшении шага результат не изменился, значит найден хороший шаг;
2. сравнить результаты двух методов разной точности (например, Рунге-Кутта 2-ого и 4-ого порядка точности);
3. запустить программу на относительно больших значениях сопротивления.

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

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

Выражения для метода трапеций:

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

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

Получили систему уравнений с двумя неизвестными: I_{n+1} и U_{n+1} . Подставим U_{n+1} из второго уравнения в первое и решим полученное уравнение относительно I_{n+1} :

$$I_{n+1} = \frac{-2C_k R_p(I_n)\Delta t + 4C_k L_k I_n - 2C_k I_n R_k \Delta t + 4C_k U_n \Delta t - I_n \Delta t^2}{4C_k L_k + 2C_k R_k \Delta t + 2C_k R_p(I_{n+1})\Delta t + \Delta t^2}.$$

Это уравнение решается методом простой итерации, то есть сначала в правую часть подставляется уже известное значение $R_p(I_{n+1})$, затем значение высчитывается повторно, но уже с новым найденным значением.

3. Из каких соображений проводится выбор численного метода того или иного порядка точности, учитывая, что чем выше порядок точности метода, тем он более сложен и требует, как правило, больших ресурсов вычислительной системы?

В случае, если функция $\phi(x, \mu)$ ограничена и непрерывна, а также ограничены и непрерывны ее N -ые производные, то оптимальный результат достигается при применении метода N -ого порядка точности. В случае, если же функция $\phi(x, \mu)$ имеет меньше N производных, то и N -ый порядок точности не сможет быть достигнут. Очевидно, тратить вычислительные ресурсы системы на расчет метода N -ого порядка точности не имеет смысла, логичнее будет использование некоторый метод меньшей точности.

4. Можно ли метод Рунге-Кутта применить для решения задачи, в которой часть условий задана на одной границе, а часть на другой? Например, напряжение по-прежнему задано при $t=0$, т.е. $U=U_0$, а ток задан в другой момент времени, к примеру, в конце импульса, т.е. при $t=T$, $I=IT$. Какой можете предложить алгоритм вычислений?

Поскольку в методе Рунге-Кутта для вычисления каждого из последующих значений необходимо знать значения обеих параметров (в нашем примере, силы тока и напряжения) в предыдущий момент времени,

то сделаю вывод, что нет, в таком случае нельзя использовать метод Рунге-Кутты.

Для решения подобной задачи необходимо использовать алгоритм, способный решать краевые задачи и уже внутри него можно использовать, например, метод Рунге-Кутты заданного порядка точности.