**МИНИСТЕРСТВО ОБРАЗОВАНИЯ РЕСПУБЛИКИ БЕЛАРУСЬ**

**УЧРЕЖДЕНИЕ ОБРАЗОВАНИЯ**

**ГОМЕЛЬСКИЙ ГОСУДАРСТВЕННЫЙ ТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ ИМЕНИ П. О. СУХОГО**

Факультет автоматизированных и информационных систем

Кафедра «Информатика»

ОТЧЕТ ПО ЛАБОРАТОРНОЙ РАБОТЕ № 5

по дисциплине «**Математическое моделирование сложных систем**»

на тему: «**Построение и исследование динамических моделей технических систем**»

Выполнил: студент гр. ИП-32

Суховенко Э.С.

Принял: доцент

Трохова Т.А.

Гомель 2022

**Цель:** получить навыки построения динамических моделей различных видов, выполнения исследований по моделям и обобщения результатов исследований с их графической интерпретацией.

**Практическая часть**

***1 часть (2 часа)***

1. С использованием СКМ рассчитать значение функций перемещения, скорости и ускорения динамической системы под воздействием начальных значений перемещения и скорости без учета возмущающей силы. Построить графики этих функций.
2. Рассчитать значение функции перемещения динамической системы под воздействием возмущающей силы. Построить графики этой функции.
3. Исследовать влияние значений изменяемого параметра на амплитуду перемещения динамической системы, для этого рассчитать функцию перемещения при различных значениях изменяемого параметра. Построить графики зависимости перемещения системы от времени.
4. Построить сводный график всех полученных функций перемещения на одном поле.
5. Построить график зависимости локального экстремума перемещения от варьируемого параметра.
6. Вычислить аналитические аппроксимирующие функции по результатам исследований предыдущего пункта. Построить графически исходные и аппроксимирующие зависимости. Сделать выводы по проведенным исследованиям.
7. ***часть (2 часа)***
8. Разработать графический пользовательский интерфейс для реализованной модели (п.1 и п.3). Обеспечить ввод варьируемого параметра модели с использованием слайдера.

8. Определить время, при котором амплитуда перемещения достигает заданного порогового значения, указанного пользователем; дать графическую интерпретацию полученным результатам.

**Задание 10. Компьютерное моделирование устройств робототехники в СКМ**

* K=35 (Нм) — коэффициент жёсткости пружины;
* KТ =0.4 (Нм/А) — моментный коэффициент двигателя.
* L=0.01 (Г) — индуктивность обмотки якоря двигателя;
* R=0.56 (Ом) — активное сопротивление обмотки якоря двигателя;
* KЕ =0.4 (Нм/Вс) — скоростной коэффициент двигателя,
* u=3B – напряжение.

**Описание математической модели**

Дифференциальное уравнение, описывающее динамику движения одного звена робота без учета влияния других звеньев, записываемое в следующем виде:

, (1)

где J — момент инерции звена;

С — коэффициент вязкого трения в подшипниках;

К — коэффициент жесткости пружины;

KТ — моментный коэффициент двигателя;

θ — угловое положение звена;

i — ток двигателя.

Динамика двигателя, управляемого от источника регулируемого напряжения, описывается следующим дифференциальным уравнением:

, (2)

где L — индуктивность обмотки якоря двигателя;

R — активное сопротивление обмотки якоря двигателя;

KЕ — скоростной коэффициент двигателя.

Схема робота представлена на рисунке 1.

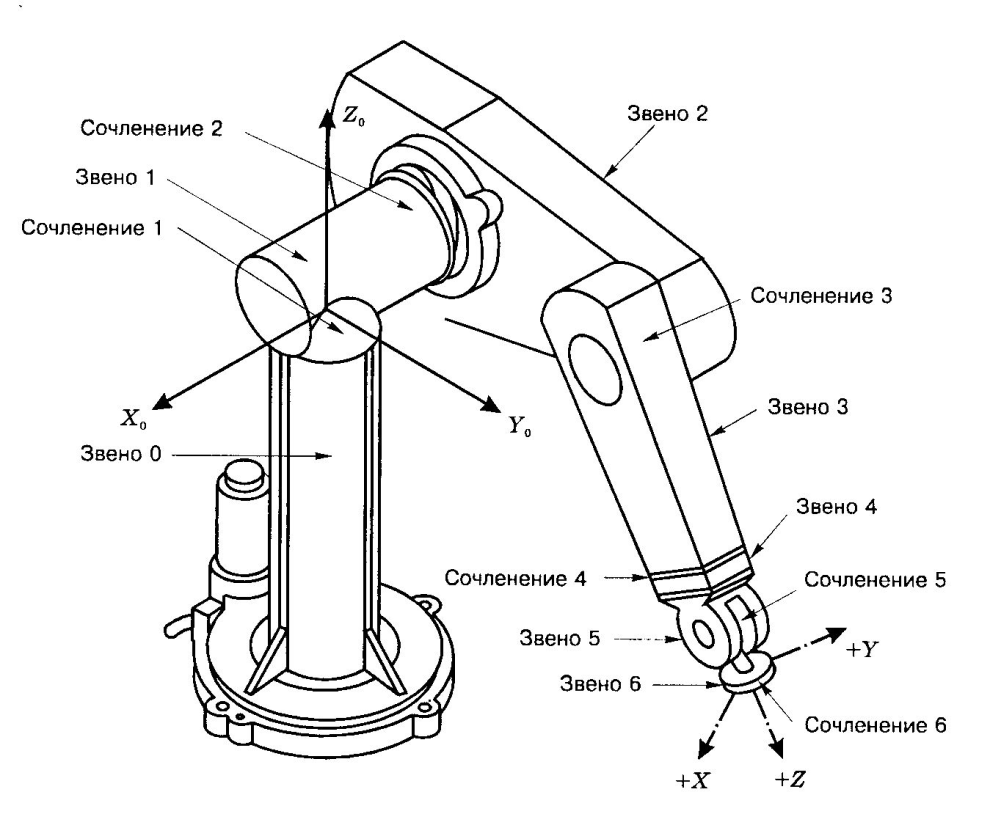


Рисунок 1 — Схема робота

**Математическая модель:**

**Листинг задания 1:**

**// main.py**

import numpy as np  
import scipy.integrate  
import matplotlib.pyplot as plt  
  
  
def create\_model(y0, t, K, Kt, L, R, Ke, u, C, J):  
 def y1\_der(y1, y2, y3):  
 return y2  
  
 def y2\_der(y1, y2, y3):  
 return 1. / J \* (-C \* y2 - K \* y1 + Kt \* y3)  
  
 def y3\_der\_u(u):  
 return lambda y1, y2, y3: 1. / L \* (u - R \* y3 - Ke \* y2)  
  
 def pend(y, t):  
 y1, y2, y3 = y  
 return [  
 y1\_der(y1, y2, y3),  
 y2\_der(y1, y2, y3),  
 y3\_der\_u(u)(y1, y2, y3)  
 ]  
  
 return scipy.integrate.odeint(pend, y0, t)  
  
  
def main():  
 K = 35 *# Nm* Kt = 0.4 *# Nm/A* L = 1 *# G* R = 0.56 *# Om* Ke = 0.4 *# Nm/(Vs)* u = 3 *# V* C = 0.3 *# Farad* J = 0.8 *# N/m* t = np.linspace(0, 10, 3000)  
 y0 = [0, 0, 0]  
  
 ls = [0.1, 0.2, 0.3, 0.4, 0.5, 0.6, 0.7, 0.8, 0.9, 1]  
 mxh = []  
 for \_l in ls:  
 model = create\_model(y0, t, K, Kt, \_l, R, Ke, u, C, J)  
 ys\_t = model[:, 0]  
 mxh.append(np.max(ys\_t))  
 plt.plot(ys\_t, linewidth=1, linestyle='-.')  
 plt.show()  
  
 min\_n = 4  
 max\_n = 4  
 for n in range(min\_n, max\_n + 1):  
 k = np.polyfit(ls, mxh, n)  
 print('Power of polynom: ', n)  
 print('Coeff approx: ', k)  
 approx\_fun = np.poly1d(k)  
 \_ls = np.linspace(0, 1, 100)  
 plt.plot(\_ls, list(map(lambda t: approx\_fun(t), \_ls)),  
 linewidth=1,  
 linestyle='--')  
 plt.plot(ls, mxh, '\*', c='red')  
 plt.grid()  
 plt.show()if \_\_name\_\_ == '\_\_main\_\_':  
 main()

**Результат выполнения работы:**

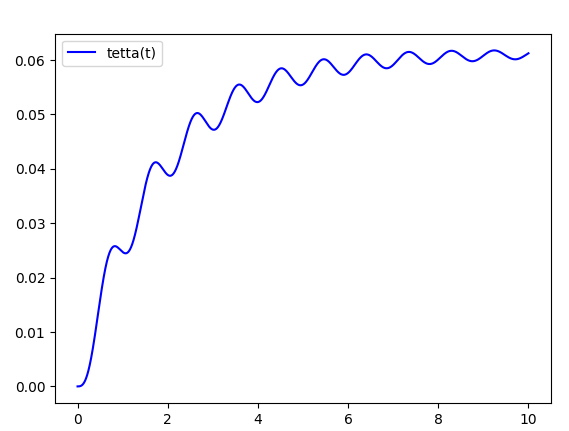
****

Рисунок 1 – График функции tetta(t)

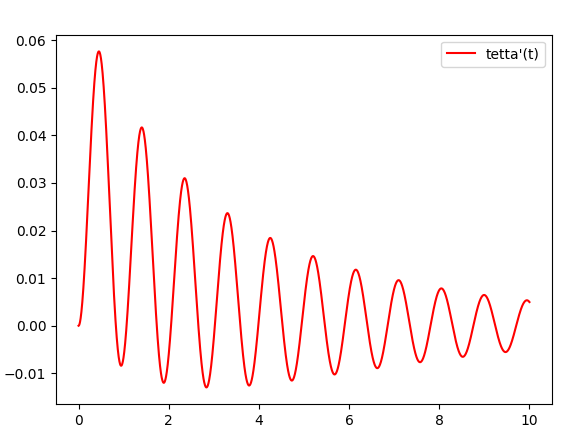
****

Рисунок 2 – График функции tetta’(t)

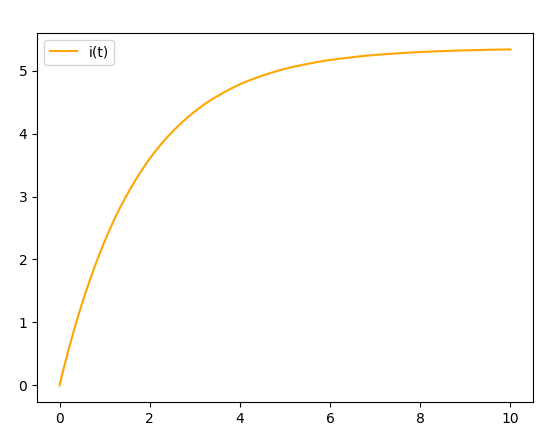
****

Рисунок 3 – График функции i(t)

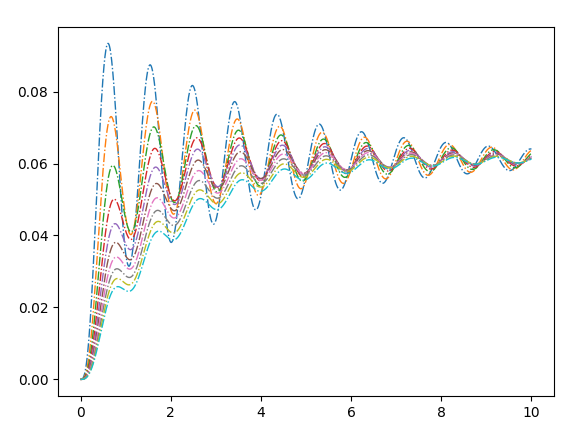
****

Рисунок 4 – График изменения параметра L

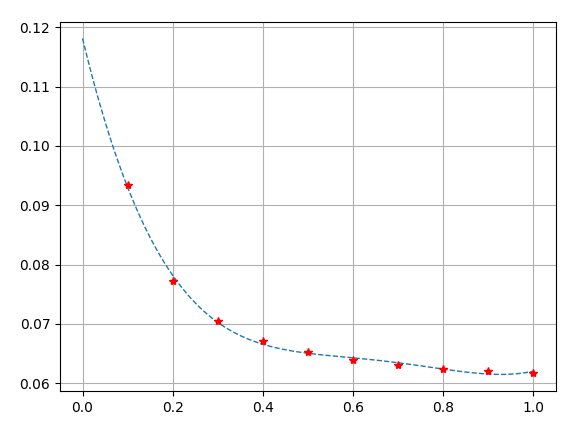


Рисунок 5 – График функции аппроксимирующей максимальные значения исходной функции при изменении параметра L

**Вывод:** получил навыки разработки планов полного факторного эксперимента в СКМ, научился выполнять регрессионный и корреляционный анализ результатов эксперимента.