**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 № 1

по дисциплине «Математические модели сложных систем»

на тему: «Построение и исследование явных аналитических моделей»

Выполнил: студент гр. ИП-31

Коваленко А.И.

Принял: преподаватель

Трохова Т.А.

Гомель 2023

**Цель:** Получить навыки компьютерного моделирования технических объектов, представленных в виде явной аналитической модели с выводом результатов моделирования в численном и графическом виде.

**Задание 1**

***1) Разработать компьютерную модель манипулятора, которая имеет следующие выходные параметры:***

- значения угла поворота звена ОА в зависимости от времени;

- значения координат шарнира А и захвата С в зависимости от времени.

Результаты моделирования представить в численном и графическом виде.

***2) Исследовать модель, для чего определить:***

***-***  максимальное значение координаты Y захвата манипулятора;

- значение координаты X, при котором координата Y захвата манипулятора максимальна;

- значение времени, при котором координата Y захвата манипулятора максимальна.

***Исходными данными для построения модели являются:***

- АВ – длина звена АВ;

- АС – длина звена АС;

- ОА – длина звена ОА;

- вид функции закона движения ползуна, заданный аналитически;

- вид функции закона движения руки АС, заданный аналитически;

- Tk – конечное значение времени для исследования модели манипулятора.

##### Таблица 1.1 - Варианты исходных данных

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| № | AB (см) | ОА (см) | АС (см) | Tk (с) | S0 | Vb | Ψ0 | ω |
| 1 | 91 | 80 | 52 | 1.2 | 1.185 | 0.862 | 0.551 | 2.247 |

**Листинг:**

**Задание 1:**

import matplotlib.pyplot as mp

import matplotlib.animation as animation

import numpy as np

Vb =0.862

S0=1.185

T=np.arange (0,1.2,0.01)

AC=0.52

OA=0.8

AB=0.91

W=2.247

Ksi0=0.551

S1 = S0 - Vb \* T

Ksil = Ksi0+W\*T

f = np.arccos((-AB\*\*2+S1\*\*2+OA\*\*2) / (2\*OA\*S1))

XA=OA\*np.cos (f)

YA=OA\*np.sin (f)

XC=XA-AC\*np.cos (f-Ksil)

YC=YA-AC\*np.sin(f-Ksil)

print (XC[np.argmax (YC) ])

print (T[np.argmax (YC) ])

print (max (YC) )

fig = mp.figure(facecolor='white')

ax = mp.axes(xlim=(-2, 2), ylim=(0, 2) )

mp.plot (XC, YC,XC [np.argmax (YC) ])

lin1, = ax.plot([ ], [ ], lw=3) # line = объект кривой

lin2, = ax.plot([ ], [ ], lw=3)

lin3, = ax.plot([ ], [ ], lw=3)

ax.grid(True)

def redraw(i):

x = XC[i]

y = YC[i]

x1=XA[i]

y1=YA[i]

x2=S1[i]

lin1.set\_data([x,x1], [y,y1])

lin2.set\_data( [0, x1], [0, y1])

lin3.set\_data([x2, x1], [0, y1])

return lin1, lin2, lin3

anim=animation.FuncAnimation(fig,redraw,frames=200,interval=30)

mp.show()

**Результат выполнения задания 1**

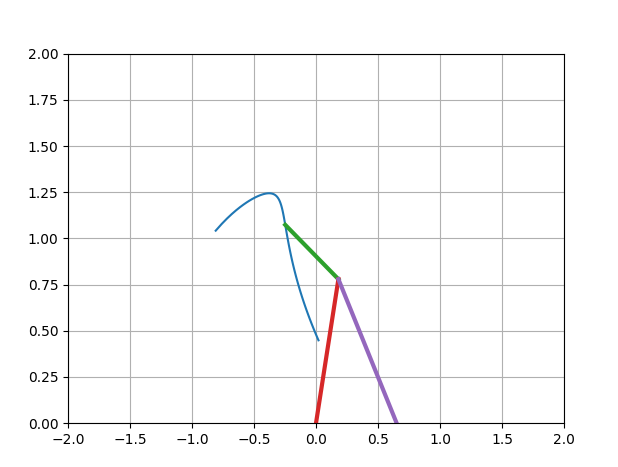
****

Рисунок 1 – Результат выполнения программы

**Задание 2:**

## Постановка задачи моделирования

***1) Разработать компьютерную модель кулачкового механизма, которая имеет следующие выходные параметры:***

- функцию аналога ускорения, скорости, перемещения толкателя в зависимости от времени;

***-*** центровой профиль кулачка.

***2) Исследовать модель, для чего определить следующие параметры:***

- максимальные и минимальные значения аналога ускорения кулачкового механизма;

- значение времени, при котором аналог ускорения кулачкового механизма максимален.

**Исходные данные:**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| №  варианта | Параметры закона S | | | | | |
|  |  |  |  |  |  |
| 4 | 1,85 | 3,14 | 5,22 | 8,0 | 6,3 | 1,256 |

**Листинг программы:**

import numpy as np

import matplotlib.pyplot as plt

v = 0.675

f1 = 1.88

f2 = 3.05

f3 = 5.12

a1 = 8.5

a2 = 7.23

W0 = 1.256

T = (2 \* np.pi) / W0

t = np.arange(0, T, 0.01)

S11 = np.zeros(len(t))

number = 0

for i in t:

if i < (f1/ W0) and i >= 0:

S11[number] = a1\*np.sin(2\*np.pi\*W0\*i/f1)

number= number+1

if i >= (f1/ W0) and i < (f2/W0) :

S11[number] = 0

number = number + 1

if i < (f2/W0 + (f3 - f2)/ W0) and i >= (f2/W0) :

S11[number] = a2\*np.sin(2\*np.pi\*W0\*i/(f3-f2))

number = number + 1

if i >= (f2/W0 + (f3 - f2)/ W0) and i < (2\*np.pi/W0) :

S11[number] = 0

number = number + 1

S1 = np.zeros(len(t))

for i in range(0, len(t), 1):

S1[i] = np.trapz(S11[0:i], t[0:i]) \* W0 \* W0

S = np.zeros(len(t))

for i in range(0, len(t), 1):

S[i] = np.trapz(S1[0:i], t[0:i])

R1 = S1 / np.tan(v)

R0 = min(R1)

R = R0 + S

X = np.zeros(len(t))

Y = np.zeros(len(t))

for i in range(0, len(t), 1):

X[i] = R[i] \* np.sin(W0 \* t[i])

Y[i] = R[i] \* np.cos(W0 \* t[i])

#1

plt.plot(t, S11, color='green')

plt.xlabel('t')

plt.ylabel('S11 ')

plt.grid(True)

plt.show()

#2

plt.plot(t, S1, color='green')

plt.xlabel('t')

plt.ylabel('S1 ')

plt.grid(True)

plt.show()

#3

plt.plot(t, S, color='green')

plt.xlabel('t')

plt.ylabel('S ')

plt.grid(True)

plt.show()

#4

plt.plot(X, Y, color='green')

plt.xlabel('X')

plt.ylabel('Y')

plt.grid(True)

plt.show()

**Результат выполнения задания 2**

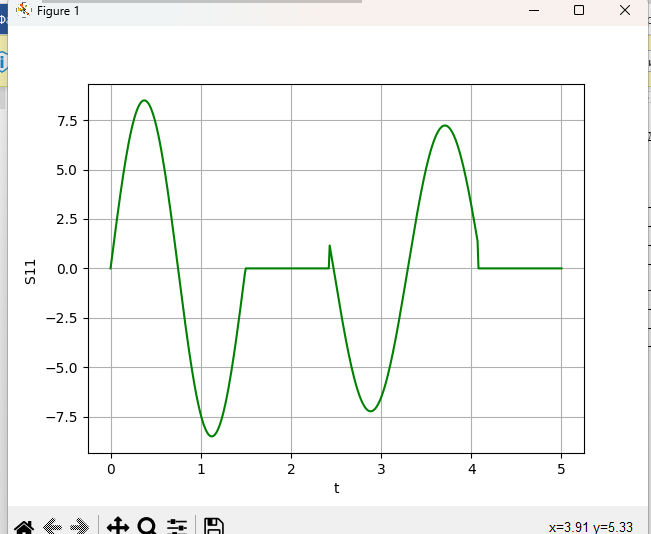
****

Рисунок 2 – Результат выполнения программы

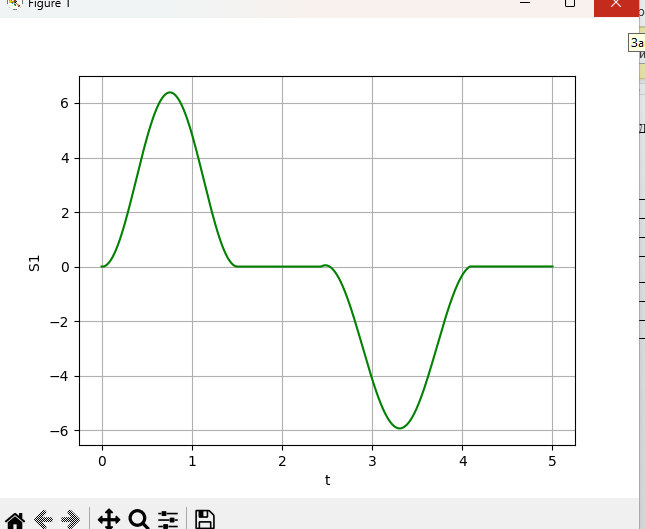
****

Рисунок 3 – Результат выполнения программы

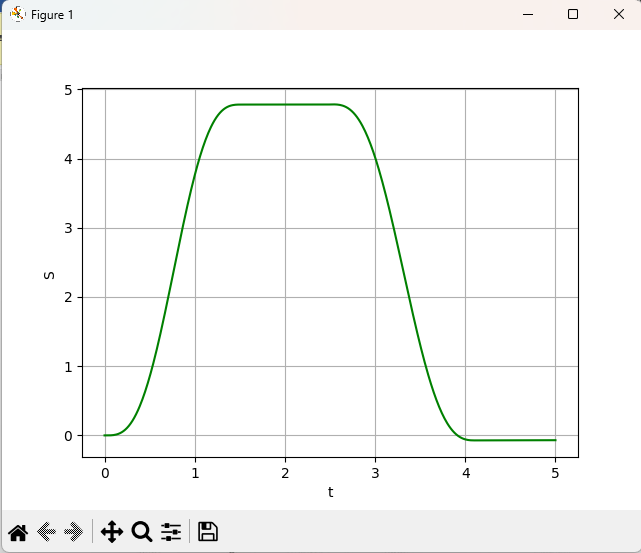
****

Рисунок 4 – Результат выполнения программы

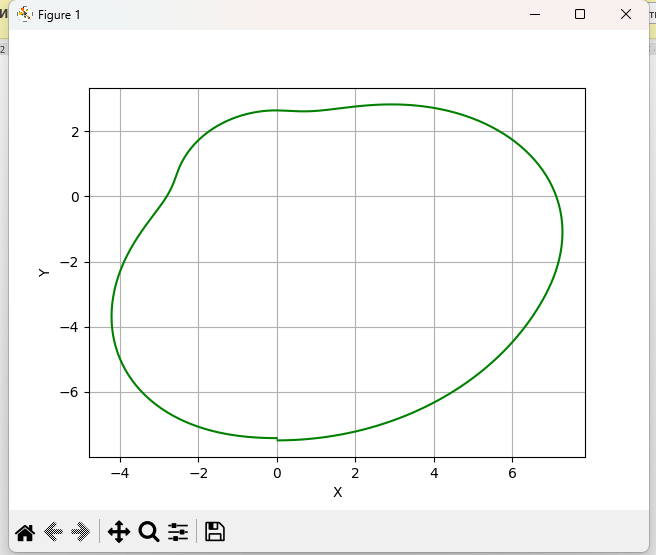
****

Рисунок 5 – Результат выполнения программы

Вывод: получили навыки компьютерного моделирования технических объектов, представленных в виде явной аналитической модели с выводом результатов моделирования в численном и графическом виде.