**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 **«Математическое моделирование сложных систем»**

на тему: **«Построение и исследование явных аналитических моделей»**

Выполнил: студент гр. ИП-31

Кузнецова Е. А.

Принял: доцент

Трохова Т. А.

Дата сдачи отчета: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Дата допуска к защите: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Дата защиты: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Гомель 2024

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

**Задание 1.**

## Постановка задачи моделирования

***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 согласно варианта определить:***

|  |  |
| --- | --- |
| № | ***Задание для исследования*** |
| 2 | * значение времени, при котором координата Y захвата манипулятора пересекает пороговое значение, равное 1.12м * значение координаты X, при котором координата Y захвата манипулятора пересекает пороговое значение, равное 1.12м |

***3) Разработать анимационную модель движения манипулятора***

***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 |
| 2 | 98 | 82 | 52 | 1.21 | 1.15 | 0.8 | 0.561 | 2.2 |

**Код программы:**

import numpy as np

import matplotlib.pyplot as plt

import matplotlib.animation as animation

AB = 0.98

OA = 0.82

AC = 0.52

Tk = 1.21

S0 = 1.15

Vb = 0.8

psy0 = 0.561

w = 2.2

t = np.linspace(0, Tk, 500)

S1 = S0 - Vb \* t #ползунок

psy = psy0 + w \* t # движение руки

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 - psy)

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

threshold\_value = 1.12

index\_threshold = np.where(YC >= threshold\_value)[0][0]

#время и координата X при пересечении порогового значения

time\_threshold = t[index\_threshold]

x\_threshold = XC[index\_threshold]

print("Время пересечения порогового значения координаты Y:", time\_threshold)

print("Координата X при пересечении порогового значения координаты Y:", x\_threshold)

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

plt.xlim(-1.5, 1)

plt.ylim(0, 1.5)

plt.plot(XC, YC)

lin1, = plt.plot([ ], [ ])

lin2, = plt.plot([ ], [ ])

lin3, = plt.plot([ ], [ ])

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])

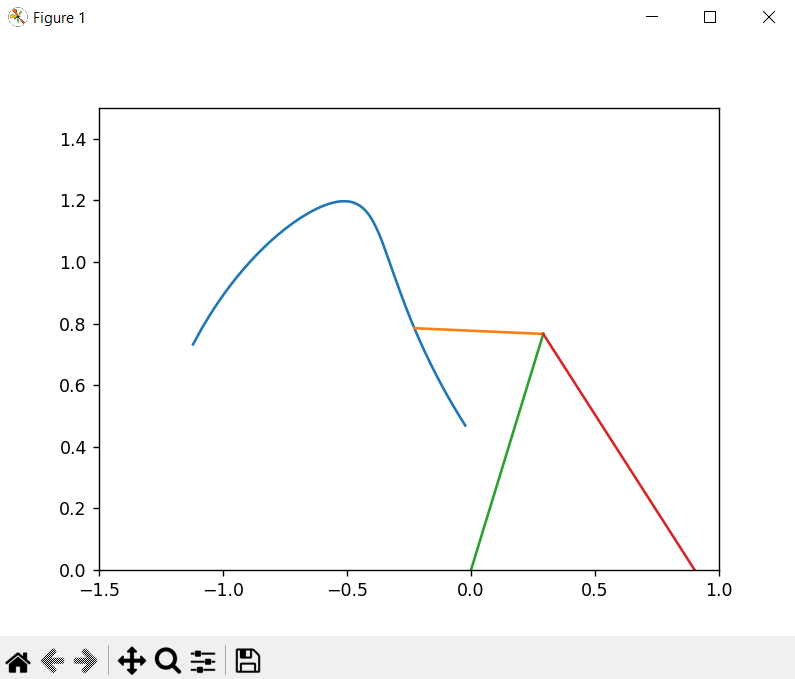
return lin1,lin2,lin3

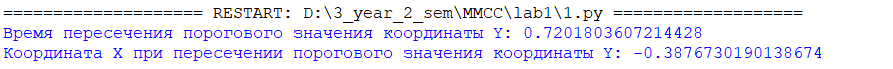
anim=animation.FuncAnimation(fig,redraw,

frames=200,interval=50)

plt.show()

**Результат выполнения:**

****

****

**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:***

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

Исходные данные для построения модели приведены в таблице 1.2.

Таблица 1.2 - Таблица исходных данных

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| №  варианта | Параметры закона S | | | | | |
|  |  |  |  |  |  |
| 2 | 1,7 | 3,2 | 5,3 | 0,86 | 0,58 | 1,256 |

Для всех вариантов v=0.675.

**Код программы:**

import numpy as np

import matplotlib.pyplot as plt

f1 = 1.7

f2 = 3.2

f3 = 5.3

a1 = 0.86

a2 = 0.58

w0 = 1.256

v = 0.675

b1 = 2 \* np.pi / f1

b2 = 2 \* np.pi / (f3 - f2)

t1 = 2 \* np.pi / (b1 \* w0)

t2 = f2 / w0

t3 = t2 + 2 \* np.pi / (b2 \* w0)

t = np.arange(0, 2 \* np.pi / w0, 0.01)

#Функция аналога ускорения роликового толкателя

def S11(t):

if 0 <= t < t1:

return a1 \* np.sin(b1 \* w0 \* t)

elif t1 <= t < t2:

return 0

elif t2 <= t < t3:

return a2 \* np.sin(b2 \* w0 \* t)

else:

return 0

#Функция скорости роликового толкателя

def S1(t):

return w0\*\*2 \* t

def R1(s1):

return s1 / np.tan(v)

#Вычисление радиуса-вектора центрового профиля кулачкового механизма

def R(s):

return R0 + s

#Центровой профиль кулачка

def XY():

X = []

Y = []

i = 0

for j in t:

X.append((R0 + s\_s1\_s11[0][i]) \* np.sin(w0 \* j))

Y.append((R0 + s\_s1\_s11[0][i]) \* np.cos(w0 \* j))

i += 1

return X, Y

#Вычисление функций

def S\_S1\_S11():

s11 = []

s1 = []

s = []

time = []

for i in t:

time.append(i)

s11.append(S11(i))

s1.append(np.trapz(list(map(S1, s11)), time))

s.append(np.trapz(s1, time))

return s, s1, s11

def graphic(x, y):

plt.plot(x, y)

plt.grid()

plt.show()

s\_s1\_s11 = S\_S1\_S11()

R0 = min(list(map(R1, s\_s1\_s11[1])))

xy = XY()

#Траектория движения толкателя

graphic(xy[0], xy[1])

#График перемещения

graphic(t, s\_s1\_s11[0])

#График скорости

graphic(t, s\_s1\_s11[1])

#График ускорения

graphic(t, s\_s1\_s11[2])

**Результат выполнения:**

Траектория движения толкателя

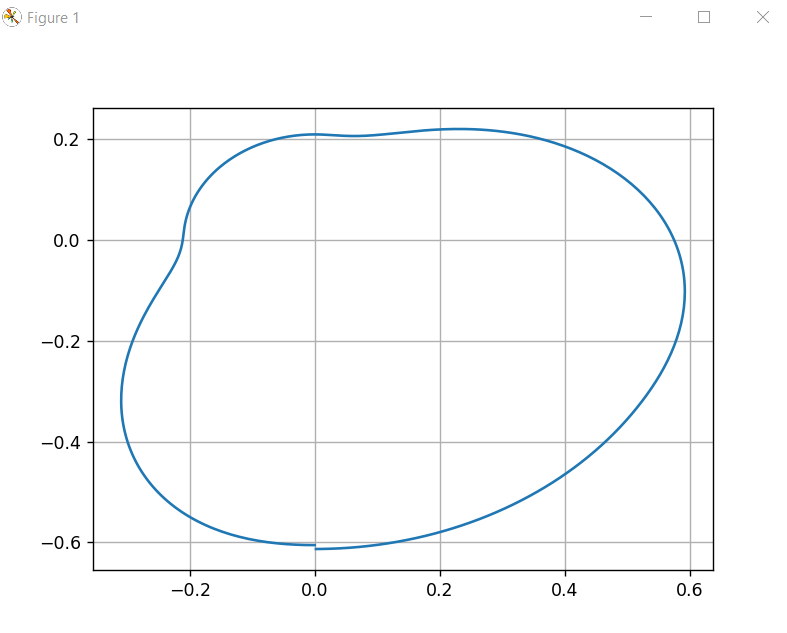
****

График перемещения

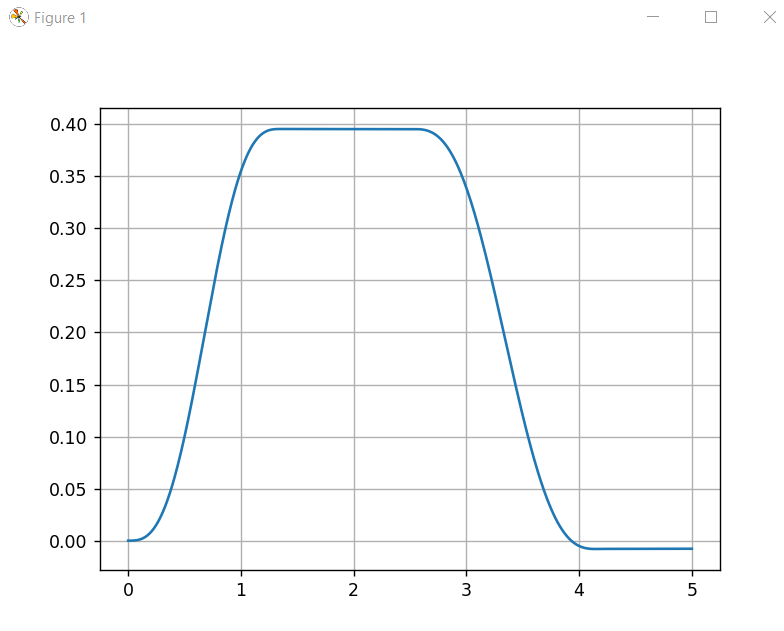


График скорости

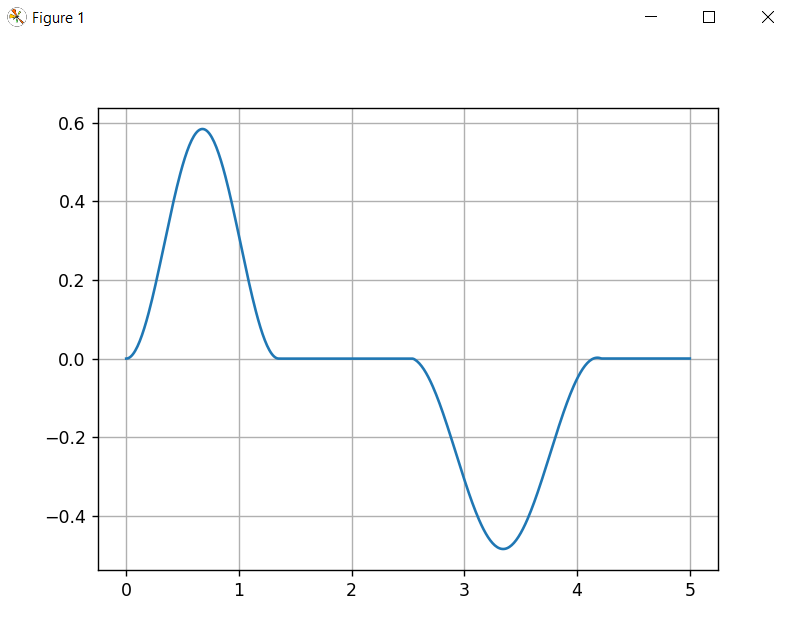
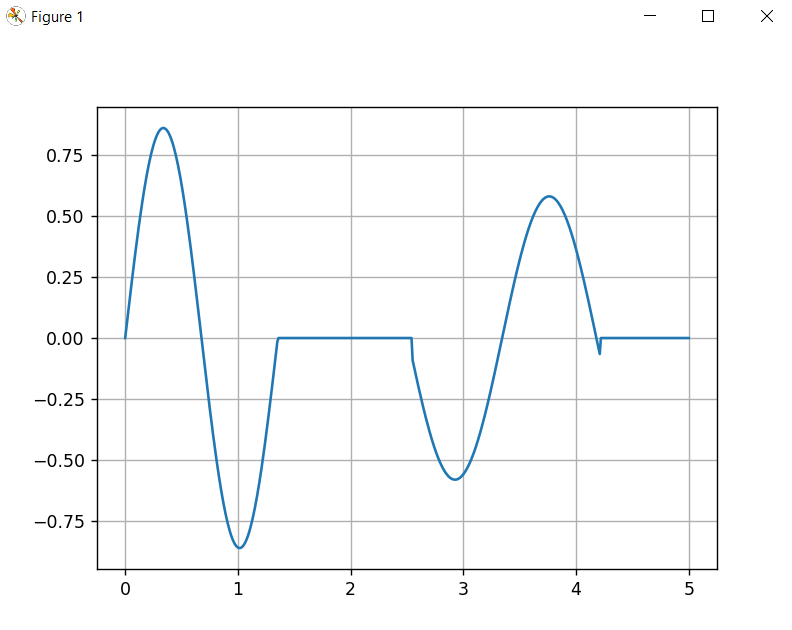


График ускорения



**Вывод:** в результате выполнения лабораторной работы мноюполучены навыки компьютерного моделирования технических объектов, представленных в виде явной аналитической модели с выводом результатов моделирования в численном и графическом виде.