**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-32

Суховенко Э.С.

Принял: доцент

Трохова Т.А.

Гомель 2022

**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 – Варианты исходных данных

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| № | AB (см) | ОА (см) | АС (см) | Tk (с) | S0 | Vb | Ψ0 | ω |
| 6 | 96 | 82 | 51 | 1.05 | 1.15 | 0.82 | 0.51 | 2.3 |

###### Описание математической модели

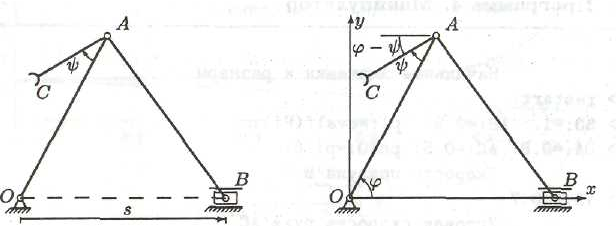


Рисунок 1 – Схема манипулятора

Механизм манипулятора (рисунок 1) приводится в движение двумя независимыми приводами. Задан закон движения ползуна:

S1(t) = S0 - Vb∙t

Закон движения руки АС относительно кривошипа ОА имеет вид:

Ψ(t) = Ψ0 + ω∙t

Координаты шарнира А вычисляются по формулам:

XA = OA∙ cos φ YA = OA∙ sin φ,

где угол φ в зависимости от времени вычисляется по формуле:



Координаты захвата вычисляются по формулам:

XC = XA - AC∙cos(φ – ψ) YC = YA - AC∙sin(φ – ψ)

**Листинг задания 1:**

import numpy as np  
import matplotlib.pyplot as plt  
import matplotlib.animation as animation  
from prettytable import PrettyTable  
  
fig = plt.figure(facecolor='white')  
  
ax = plt.axes(xlim=(-1, 2), ylim=(0, 1.5))  
line, = ax.plot([], [], lw=3)  
lin1, = ax.plot([], [])  
lin2, = ax.plot([], [])  
lin3, = ax.plot([], [])  
ax.grid(True)  
  
AB = 0.96  
OA = 0.82  
AC = 0.51  
S0 = 1.15  
Vb = 0.82  
YY0 = 0.51  
omega = 2.3  
  
t = np.linspace(0, 1.05, 100)  
S1 = S0 - Vb \* t  
YY = YY0 + omega \* t  
Fi = np.arccos(((-AB \*\* 2) + (S1 \*\* 2) + (OA \*\* 2)) / (2 \* OA \* S1))  
  
XA = OA \* np.cos(Fi)  
YA = OA \* np.sin(Fi)  
  
XC = XA - AC \* np.cos(Fi - YY)  
YC = YA - AC \* np.sin(Fi - YY)  
  
*# table*my\_table = PrettyTable()  
my\_table.field\_names = ["№", "t", "fi", "ax", "ay", "cx", "cy"]  
for i in range(len(t)):  
 my\_table.add\_row((i, t[i], Fi[i], XA[i], YA[i], XC[i], YC[i]))  
print(my\_table)  
  
ymax = np.max(YC)  
ind = np.where(YC == ymax)  
xmax = XC[ind]  
  
print('y max = ', ymax)  
print('x max = ', xmax)  
print('t = ', t[ind])  
  
  
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  
  
  
plt.plot(XC, YC)  
anim = animation.FuncAnimation(fig, redraw, frames=100, interval=50)  
plt.show()

**Результат выполнения задания 1:**

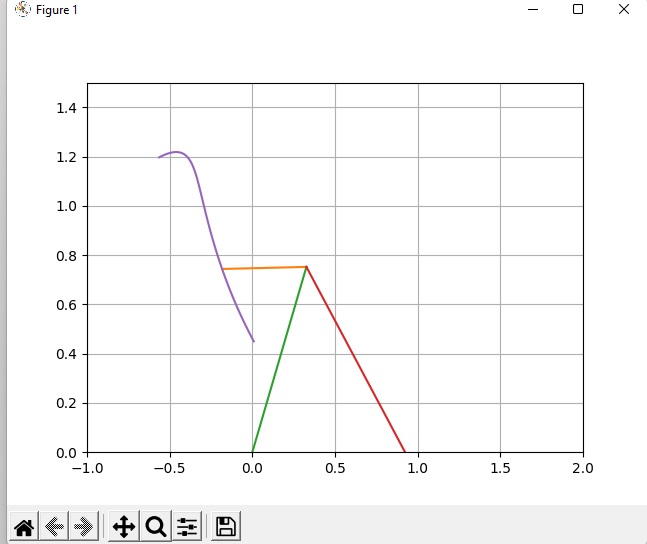
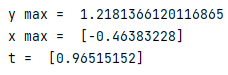


Рисунок 2 – Результат выполнения задания 1

**Задание 2:**

***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.

Таблица 2 – Таблица исходных данных

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| №  варианта | Параметры закона S | | | | | |
|  |  |  |  |  |  |
| 2 | 1,7 | 3,2 | 5,3 | 8,6 | 5,8 | 1,256 |

Для всех вариантов ***v***=0.675

###### Описание математической модели

Дан кулачковый механизм, закон изменения аналога ускорения толкателя которого приведен на рисунке 3.

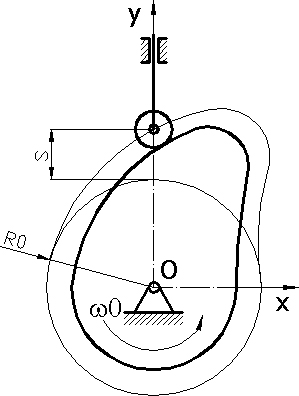


Рисунок 3 – Кулачковый механизм

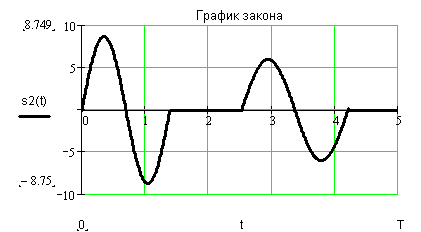


Рисунок 4 – Функция аналога ускорения роликового толкателя

Функция аналога ускорения роликового толкателя имеет вид:





Функция скорости роликового толкателя имеет вид:

S1(t) =



Функция перемещения роликового толкателя имеет вид:

S(t) =



Для вычисления радиуса-вектора центрового профиля кулачкового механизма применяются следующая зависимость:

R=R0 + S(t)

Здесь R0 вычисляется как минимальное значение функции R1(t) закона изменения радиуса-вектора центрового профиля, удовлетворяющего допустимому значению угла давления.



Центровой профиль кулачка определяется системой параметрических уравнений в декартовой системе координат:

*X(t)=R(t)sin(w0⋅t)*

*Y(t)=R(t)cos(w0⋅t)*

**Листинг задания 2:**

import numpy as np  
import matplotlib.pyplot as plt  
import math  
  
phi1 = 1.87  
phi2 = 3.05  
phi3 = 5.12  
a1 = 8.5  
a2 = 7.0  
omg0 = 1.256  
nu = 0.675  
  
b1 = 2 \* math.pi / phi1  
b2 = 2 \* math.pi / (phi3 - phi2)  
  
t1 = 2 \* math.pi / (b1 \* omg0)  
t2 = phi2 / omg0  
t3 = t2 + 2 \* math.pi / (b2 \* omg0)  
  
delta = 0.01  
  
  
def s11\_f(t):  
 if 0 <= t < t1:  
 return a1 \* math.sin(b1 \* omg0 \* t)  
 elif t1 <= t < t2:  
 return 0  
 elif t2 <= t < t3:  
 return a2 \* math.sin(b2 \* omg0 \* t)  
 elif t3 <= t < (2 \* np.pi) / omg0:  
 return 0  
  
  
def s1\_f(t, a):  
 temp = []  
 for x in t:  
 temp.append(omg0 \*\* 2 \* x)  
 return np.trapz(temp, a)  
  
  
def s\_f(t, a):  
 return np.trapz(t, a)  
  
  
def r0\_func(s1):  
 min = s1[0] / math.tan(nu)  
 for x in s1:  
 if x / math.tan(nu) < min:  
 min = x / math.tan(nu)  
 return min  
  
  
def x\_f(i, t, R, omg0):  
 return R[i] \* math.cos(t \* omg0)  
  
  
def y\_f(i, t, R, omg0):  
 return R[i] \* math.sin(t \* omg0)  
  
  
if \_\_name\_\_ == '\_\_main\_\_':  
 a = 2 \* np.pi / omg0  
 dif = a - 0  
 t = np.arange(0, a, dif / 500)  
 s = []  
 s1 = []  
 s11 = []  
 temp = []  
 res = []  
  
 *# integral* for x in t:  
 temp.append(x)  
 s11.append(s11\_f(x))  
 s1.append(s1\_f(s11, temp))  
 s.append(s\_f(s1, temp))  
 res.append(s\_f(s1, temp))  
  
 R0 = r0\_func(s1)  
 R = []  
 for x in s:  
 R.append(R0 + x)  
  
 x = []  
 y = []  
 i = 0  
 for z in t:  
 x.append(x\_f(i, z, R, omg0))  
 y.append(y\_f(i, z, R, omg0))  
 i += 1  
  
 plt.plot(y, x, color='orange')  
 plt.show()

**Результат выполнения задания 2:**

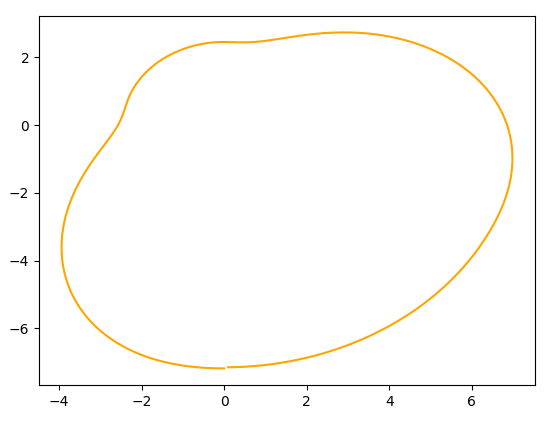


Рисунок 5 – Результат выполнения задания 2

**Вывод:** в ходе данной лабораторной работы были получены навыки компьютерного моделирования технических объектов, представленных в виде явной аналитической модели с выводом результатов моделирования в численном и графическом виде.