**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 № 3

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

на тему: «**Построение моделей по экспериментальным данным**»

Выполнил: студент гр. ИП-31

Коваленко А.И.

Принял: доцент

Трохова Т.А.

Гомель 2023

**Цель:** получить навыки разработки моделей по результатам эксперимента, применить функции аппроксимации, выполнить исследования по моделям.

**Задание 1:**

**Постановка задачи моделирования**

1. Ввести координаты опорных точек для фиксации обязательного захода в них робота.

2. Подобрать аппроксимирующую аналитическую функцию для вычисления зависимости Y(X) движения мобильного робота в плоскости.

3. Построить график движения мобильного робота по рассчитанной траектории.

4. Найти максимальное отклонение по координате Y полученной траектории движения от опорной точки.

5. Найти расстояние, пройденное роботом от исходной точки до конечной, если он движется по траектории Y(X).

|  |  |  |  |
| --- | --- | --- | --- |
| 3. |  | 4;4.3;4.6 ;4.9; 5.2; 5.5; 5.8 | -3.2;6;-1.3;4;0.1;3.3;-2.1 |

**Листинг задания 1:**

import scipy.misc

import numpy as np

from matplotlib import pyplot as plt

from scipy.optimize import curve\_fit

from scipy import integrate

xs = [3.9, 4.23, 4.6, 4.9, 5.1, 5.5, 5.8]

ys = [-16, -7, 4, 19, 37, 60, 87]

def y\_f(x, a, b):

return a \* x\*\*3 + b \* x\*\*2

k, s = curve\_fit(y\_f, xs, ys, (0, 0))

[a, b] = k

axs = np.arange(xs[0], xs[-1], 0.01)

ays = list(map(lambda x: y\_f(x, a, b), axs))

get\_delt = lambda a, b: np.abs(a - b)

def len\_f(x):

return np.sqrt(1 + scipy.misc.derivative(lambda x: y\_f(x, a, b), x, dx=1.0, n=1) \*\* 2)

full\_dist = integrate.quad(len\_f, xs[0], xs[-1])

print('Полная дистанция, погрешность', full\_dist)

print("Максимальное отклонение по координате Y", xs[0], xs[-1])

print("k=", k)

plt.plot(axs, ays)

plt.plot(xs, ys, '+')

plt.grid()

plt.show()

**Результат выполнения задания 1:**

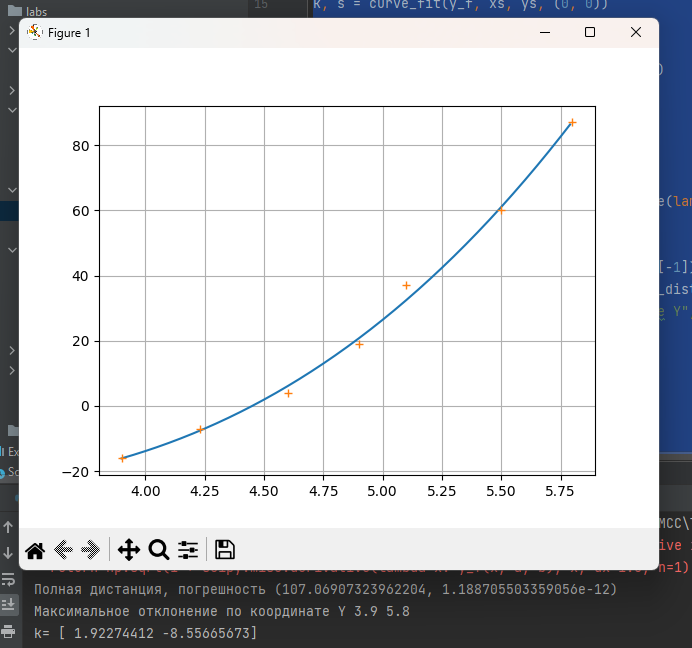


Рисунок 1 – Результат выполнения задания 1

**Задание 2:**

**Постановка задачи моделирования**

1. Ввести координаты опорных точек вольт-амперной характеристики, используя график приложения А.

2. Подобрать аппроксимирующую аналитическую функцию I(u) для заданной графически вольт-амперной характеристики.

3. Построить график исходной и аппроксимирующей характеристик.

**Описание математической модели**

Электрическая цепь, приведенная на рисунке 2, включает туннельный диод с нелинейной вольт-амперной характеристикой. Упрощенная модель диода имеет вид:



Начальные значения для аппроксимации функции вольт-амперной характеристики диода выбрать следующие:

А=0.2

α=7

D=10-7

β =10

*R*

*i(t)*

*u*

*E*

*C*

*I(u)*

Рисунок 1

*L*

N

**Листинг задания 2:**

import numpy as np

from scipy.optimize import curve\_fit

import matplotlib.pyplot as plt

def func(u, A, D, alpha, beta):

return A \* u \* np.exp(-alpha \* u) + D \* (np.exp(beta \* u) - 1)

A = 0.2

alpha = 7

D = 1e-7

beta = 10

Iu = np.array([0.003, 0.006, 0.009, 0.012, 0.009, 0.008, 0.006, 0.004, 0.003, 0.002, 0.003, 0.005])

u = np.array([0.01, 0.03, 0.06, 0.11, 0.24, 0.28, 0.35, 0.42, 0.5, 0.56, 0.63, 0.7])

Res, pog = curve\_fit(func, u, Iu, (A, D, alpha, beta))

A1 = Res[0]

D1 = Res[1]

alpha1 = Res[2]

beta1 = Res[3]

u1 = np.linspace(0, 0.7, 100)

Iu1 = func(u1, A1, D1, alpha1, beta1)

plt.scatter(u, Iu, color='orange')

plt.plot(u1, Iu1)

plt.show()

**Результат выполнения задания 2:**

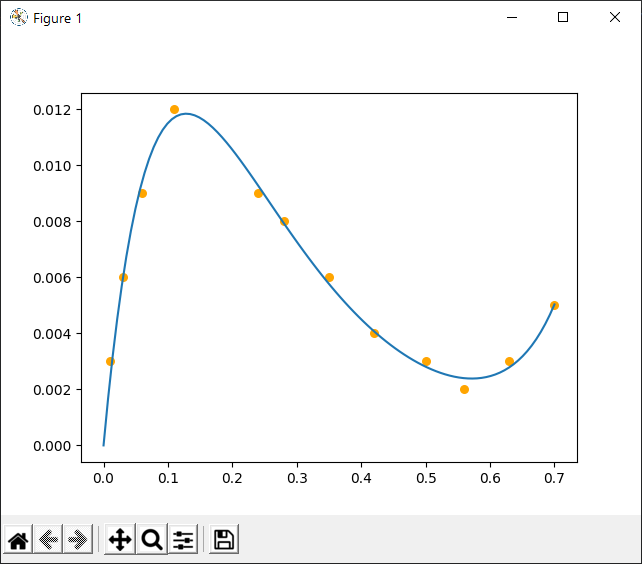


Рисунок 3 – Результат выполнения задания 2

**Вывод:** в ходе данной лабораторной работы были получены навыки разработки моделей по результатам эксперимента, применены функции аппроксимации, выполнены исследования по моделям.