

Отчет по лабораторной работе 5

НФИбд-02-18

Оразклычев Довлет

Содержание

1	Цель работы	5
2	Задание	6
3	Выполнение лабораторной работы	7
4	Вывод	14

List of Tables

List of Figures

2.1	Задание лабораторной работы	6
3.1	График зависимости численности хищников от численности жертв	9
3.2	Стационарное состояние системы	10

1 Цель работы

Построить график зависимости численности хищников от численности жертв

2 Задание

$$\begin{cases} \frac{dx}{dt} = -0.17x(t) + 0.046x(t)y(t) \\ \frac{dy}{dt} = 0.37y(t) - 0.034x(t)y(t) \end{cases}$$

Постройте график зависимости численности хищников от численности жертв, а также графики изменения численности хищников и численности жертв при следующих начальных условиях: $x_0 = 11$, $y_0 = 16$. Найдите стационарное состояние системы.

Figure 2.1: Задание лабораторной работы

3 Выполнение лабораторной работы

Для начала мы импортируем библиотеки для построения кода и вводим наши переменные:

```
import numpy as np
import matplotlib.pyplot as plt

from scipy.integrate import odeint

a = 0.17
b = 0.046
c = 0.37
d = 0.034

t0 = 0
tmax = 400
dt = 0.1
```

Теперь мы создаем список значений t , которое мы будем использовать чтобы вычислять поточечно значения “Численность армии”:

```
t = np.arange(t0, tmax, dt)
t = np.append(t, tmax)
```

Обратите внимание, что я также добавил элемент t_{\max} в конец списка. Дело в том, что функция `np.arange` заполняет от нуля до $t_{\max} - dt$, поэтому надо

добавлять еще один элемент отдельно.

Теперь создаем систему уравнений:

```
def syst(x, t):  
    return x[1], -w * w * x[0] - g * x[1] - p(t)
```

Создаем вектор значений наших данных и запускаем команду odeint, которая найдет значения поточечно.

```
v0 = (11, 16)
```

```
yf = odeint(syst, v0, t)
```

```
x = []
```

```
y = []
```

```
for i in range(len(yf)):  
    x.append(yf[i][0])  
    y.append(yf[i][1])
```

Теперь создаем график и выводим на экран. график будет красного цвета с обозначением “x”. Размер графика 10 на 10 единиц.

```
plt.figure(figsize=(10, 10))  
plt.plot(x, y, 'r', label='x')  
plt.show()
```

И получаем:

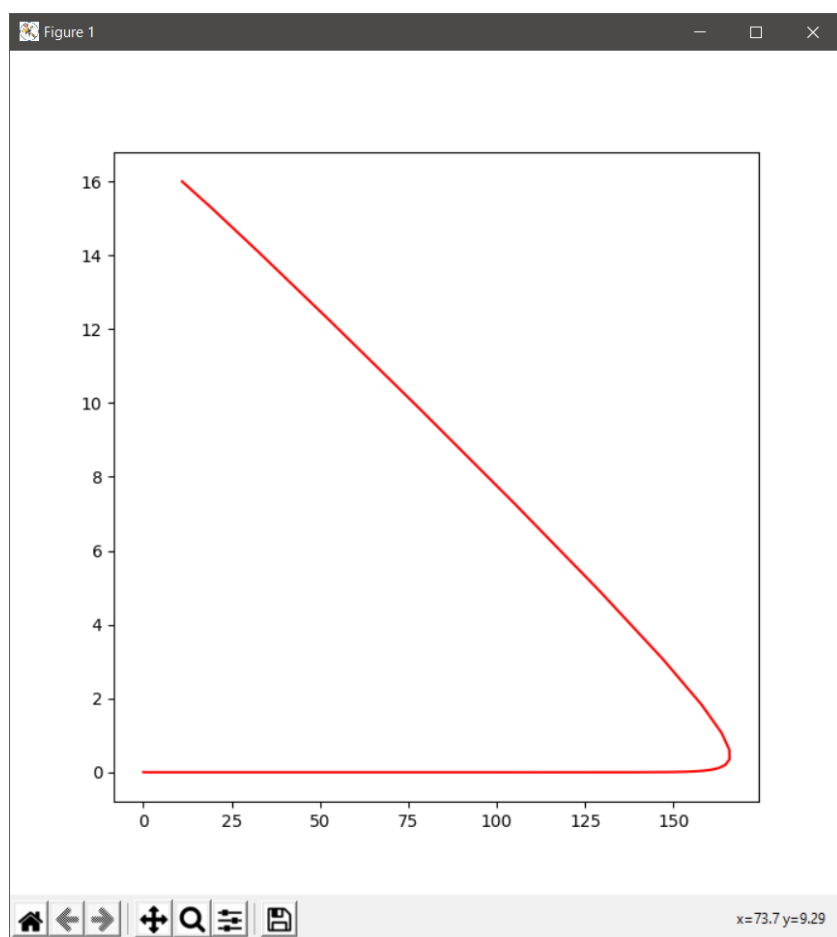


Figure 3.1: График зависимости численности хищников от численности жертв

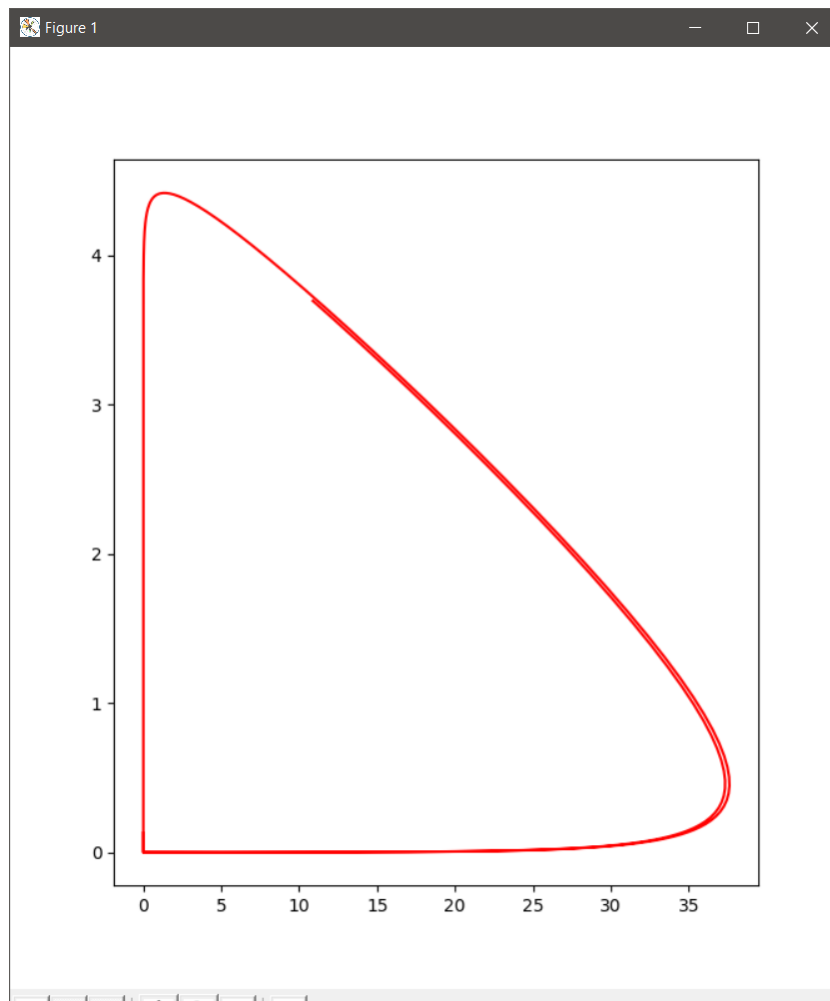


Figure 3.2: Стационарное состояние системы

Код на Python для графика 1:

```
import numpy as np
import matplotlib.pyplot as plt

from scipy.integrate import odeint

a = 0.17
b = 0.046
c = 0.37
d = 0.034
```

```

t0 = 0
tmax = 400
dt = 0.1

t = np.arange(t0, tmax, dt)
t = np.append(t, tmax)

def syst(x, t):
    dx1 = -a * x[0] + c * x[0] * x[1]
    dx2 = b * x[1] - d * x[0] * x[1]
    return dx1, dx2

v0 = (11, 16)

yf = odeint(syst, v0, t)

x = []
y = []

for i in range(len(yf)):
    x.append(yf[i][0])
    y.append(yf[i][1])

plt.figure(figsize=(10, 10))
plt.plot(x, y, 'r', label='x')
plt.show()

```

Код на Python для графика 2:

```
import numpy as np
import matplotlib.pyplot as plt

from scipy.integrate import odeint

a = 0.17
b = 0.046
c = 0.37
d = 0.034

t0 = 0
tmax = 400
dt = 0.1

t = np.arange(t0, tmax, dt)
t = np.append(t, tmax)

def syst(x, t):
    dx1 = -a * x[0] + c * x[0] * x[1]
    dx2 = b * x[1] - d * x[0] * x[1]
    return dx1, dx2

v0 = (c/d, a/b)
```

```
yf = odeint(syst, v0, t)

x = []
y = []

for i in range(len(yf)):
    x.append(yf[i][0])
    y.append(yf[i][1])

plt.figure(figsize=(10, 10))
plt.plot(x, y, 'r', label='x')
plt.show()
```

4 Вывод

Построили код на Python для решения и вывода на экран графика зависимости численности хищников от численности жертв, а также стационарного состояния системы.