

Отчет для лабораторной работы №3

НФИбд-02-18

Оразклычев Давут

Содержание

1	Цель работы	5
2	Задание	6
3	Выполнение лабораторной работы	7
4	Выводы	16

List of Tables

List of Figures

2.1	Задание	6
3.1	Модель боевых действий между регулярными войсками	10
3.2	Модель ведение боевых действий с участием регулярных войск и партизанских отрядов	11

1 Цель работы

Решение заданий

2 Задание

Между страной X и страной Y идет война. Численность состава войск исчисляется от начала войны, и являются временными функциями $x(t)$ и $y(t)$. В начальный момент времени страна X имеет армию численностью 32500 человек, а в распоряжении страны Y армия численностью в 13800 человек. Для упрощения модели считаем, что коэффициенты a, b, c, h постоянны. Также считаем $P(t)$ и $Q(t)$ непрерывные функции.

Постройте графики изменения численности войск армии X и армии Y для следующих случаев: (рис. 2.1)

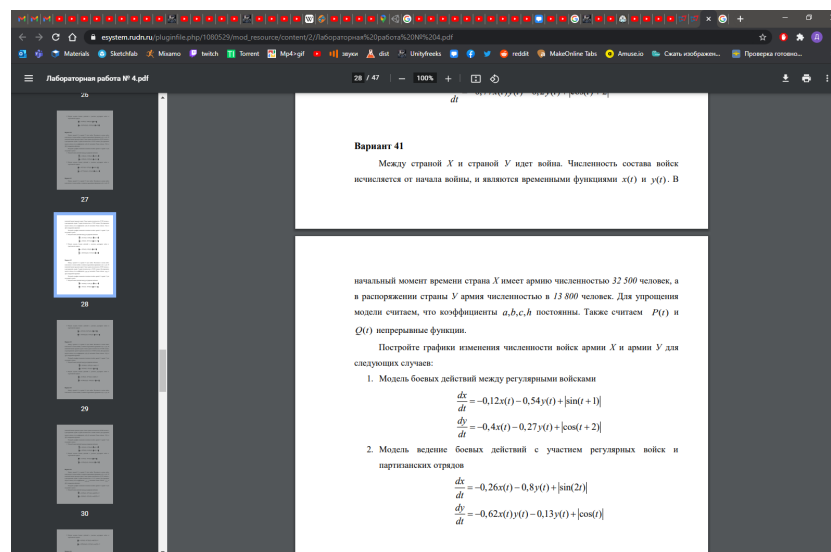


Figure 2.1: Задание

3 Выполнение лабораторной работы

Импортируем библиотеки и переменные

```
import matplotlib.pyplot as plt
import math
import numpy as np
import pandas
import scipy as sp
from scipy.integrate import odeint
X_p = 32500
Y_p = 13800
T_p = 0
///<Задание 1
a = 0.12
b = 0.54
c = 0.4
h = 0.27
///
///<Задание 2
a = 0.26
b = 0.8
c = 0.62
h = 0.13
///
```

```
Limit_time = 3
Step_p = 0.05
```

Создаем список значений Time_p, для вычислять поточечно значения 'Численность состава войск':

```
Time_p = np.arange(T_p, Limit_time, Step_p)
Time_p = np.append(Time_p, Limit_time)
```

Создаем функции и уравнение:

--функция p

```
def p(Time_p):
    return abs((math.sin(Time_p+1)))
```

--функция q

```
def q(Time_p):
    return abs((math.cos(Time_p+2)))
```

--уравнение

```
def Equation_p(Function_p, Time_p):
    x = Function_p[0]
    y = Function_p[1]
    dxdt = -a*x - b*y + p(Time_p)
    ///Задание 1
    dydt = -c*x - h*y + q(Time_p)
    ///
    ///Задание 2
    dydt = -c*x*y - h*y + q(Time_p)
    ///
    return (dxdt, dydt)
```

Создаем вектор значений


```
Vector_p = (X_p, Y_p)
```

```
//даем ему наши функции odeint
```

```
Answer_p = odeint(Equation_p, Vector_p, Time_p)
```

```
//создаем заранее списки
```

```
x = []
```

```
y = []
```

```
//через цикл добавляем ответы в массив X и массив Y
```

```
for i in range(len(Answer_p)):
```

```
    x.append(Answer_p[i][0])
```

```
    y.append(Answer_p[i][1])
```

```
//создаем нулевую полосу где по графику мы поймем что война идет в чьюто
```

```
zero = []
```

```
for i in range (len(Time_p)):
```

```
    zero = np.append(zero,0)
```

Теперь создаем график и показываем результаты на дисплее

```
//создаем график
```

```
plt.figure(figsize = (20, 15))
```

```
//полоска нулевая будет синей
```

```
plt.plot(Time_p, zero, 'b')
```

```
//пишем кривую которая посторется по точкам которые находятся по координ
```

```
plt.plot(Time_p, x, 'r', label = 'x')
```

```
//тоже самое только по списку Y
```

```
plt.plot(Time_p, y, 'g', label = 'y')
```

```
plt.ylabel('Численность состава войск')
```

```

plt.xlabel('Время')
///Задание 1
plt.title('Модель боевых действий между регулярными войсками')
///

///Задание 2
plt.title('Модель ведение боевых действий с участием регулярных войск и г
///
plt.legend(loc='upper right')

plt.show()

```

И получаем: (рис. 3.1)

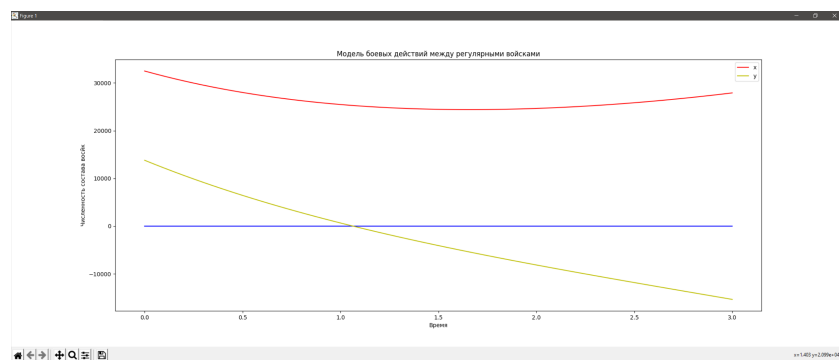


Figure 3.1: Модель боевых действий между регулярными войсками

(рис. 3.1)

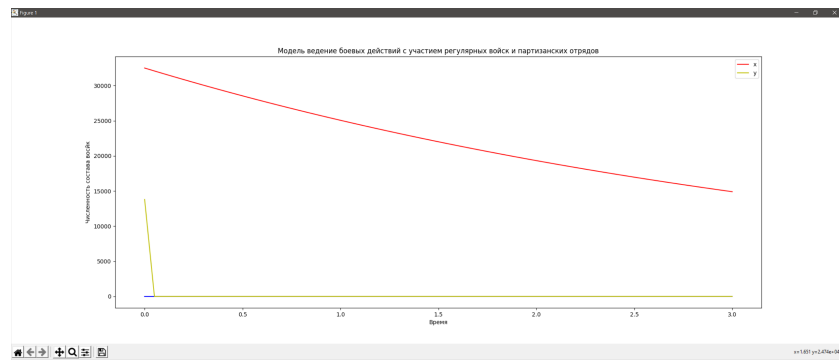


Figure 3.2: Модель ведение боевых действий с участием регулярных войск и партизанских отрядов

Код на Python для случая 1:

```
import matplotlib.pyplot as plt
import math
import numpy as np
import pandas
import scipy as sp
from scipy.integrate import odeint
```

```
X_p = 32500
```

```
Y_p = 13800
```

```
T_p = 0
```

```
a = 0.12
```

```
b = 0.54
```

```
c = 0.4
```

```
h = 0.27
```

```
Limit_time = 3
```

```
Step_p = 0.05
```

```
Time_p = np.arange(T_p, Limit_time, Step_p)
```

```
Time_p = np.append(Time_p, Limit_time)
```

```
def p(Time_p):  
    return abs((math.sin(Time_p+1)))
```

```
def q(Time_p):  
    return abs((math.cos(Time_p+2)))
```

```
def Equation_p(Function_p,Time_p):  
    x = Function_p[0]  
    y = Function_p[1]  
    dxdt = -a*x-b*y+p(Time_p)  
    dydt = -c*x - h*y+q(Time_p)  
    return (dxdt,dydt)
```

```
Vector_p = (X_p,Y_p)
```

```
Answer_p = odeint(Equation_p,Vector_p,Time_p)
```

```
x = []
```

```
y = []
```

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

```

zero = []
for i in range (len(Time_p)):
    zero = np.append(zero,0)

plt.figure(figsize =(20,15))
plt.plot(Time_p,zero, 'b')
plt.plot(Time_p,x,'r',label = 'x')
plt.plot(Time_p,y,'y',label = 'y')
plt.ylabel('Численность состава войск')
plt.xlabel('Время')
plt.title('Модель боевых действий между регулярными войсками')
plt.legend(loc ='upper right')

plt.show()

```

Код на Python для случая 2:

```

import matplotlib.pyplot as plt
import math
import numpy as np
import pandas
import scipy as sp
from scipy.integrate import odeint

X_p = 32500

Y_p = 13800
T_p = 0

```

```

a = 0.26
b = 0.8
c = 0.62
h = 0.13

Limit_time = 3
Step_p = 0.05

Time_p = np.arange(T_p, Limit_time, Step_p)

Time_p = np.append(Time_p, Limit_time)

def p(Time_p):
    return abs((math.sin(Time_p+1)))

def q(Time_p):
    return abs((math.cos(Time_p+2)))

def Equation_p(Function_p,Time_p):
    x = Function_p[0]
    y = Function_p[1]
    dxdt = -a*x-b*y+p(Time_p)
    dydt = -c*x*y - h*y+q(Time_p)
    return (dxdt,dydt)

Vector_p = (X_p,Y_p)

Answer_p = odeint(Equation_p,Vector_p,Time_p)

```

```

x = []
y = []

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

zero = []
for i in range (len(Time_p)):
    zero = np.append(zero,0)

plt.figure(figsize =(20,15))
plt.plot(Time_p,zero, 'b')
plt.plot(Time_p,x,'r',label = 'x')
plt.plot(Time_p,y,'y',label = 'y')
plt.ylabel('Численность состава войск')
plt.xlabel('Время')
plt.title('Модель ведение боевых действий с участием регулярных войск и г
plt.legend(loc ='upper right')

plt.show()

```

4 Выводы

Написали скрипт для вывода моделей боевых действий на дисплей.