Лабораторная работа №3

Модель боевых действий

Дидусь К.В.

Содержание

# Цель работы

* Рассмотреть простейшую модель боевых действий - модель Ланчестера:

# Задание

1. Составить системы дифференциальных уравнений изменения численностей армий;
2. Построить график развития боевых действий между двумя регулярными армиями;
3. Построить график развития боевых действий между регулярной армией и отрядами партизан;

## Условие

Между страной Х и страной У идет война. Численность состава войск исчисляется от начала войны, и являются временными функциями и . В начальный момент времени страна Х имеет армию численностью 12 000 человек, а в распоряжении страны У армия численностью в 15 000 человек. Для упрощения модели считаем, что коэффициенты постоянны. Также считаем и непрерывные функции.

Постройте графики изменения численности войск армии Х и армии У для следующих случаев:

Между регулярными войсками:

Между регулярными и партизанами:

# Выполнение лабораторной работы

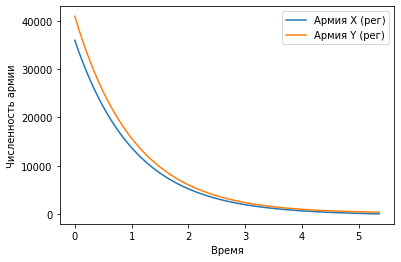
## Код программы

Приведу полный код программы (Python):

import numpy as np  
from scipy.integrate import odeint  
import matplotlib.pyplot as plt  
  
x0 = 36010 #численность армии Х  
y0 = 41000 #численность армии Y  
  
# Между регулярными:  
a1 = 0.12 #константа, характеризующая степень влияния различных факторов на потери  
b1 = 0.74 #эффективность боевых действий армии у  
c1 = 0.49 #эффективность боевых действий армии х  
h1 = 0.53 #константа, характеризующая степень влияния различных факторов на потери  
  
# Между регулярными и партизанами:  
a2 = 0.47 #константа, характеризующая степень влияния различных факторов на потери  
b2 = 0.654 #эффективность боевых действий армии у  
c2 = 0.456 #эффективность боевых действий армии х  
h2 = 0.39 #константа, характеризующая степень влияния различных факторов на потери  
  
# время  
t = np.arange(0,0.0001, 0.0000001)  
  
# Первый случай  
def P1(t):  
 p1 = np.sin(t+1) + 2  
 return p1  
def Q1(t):  
 q1 = np.cos(t+2) + 2  
 return q1  
  
# Второй случай  
def P2(t):  
 p2 = 1.5 \* abs(np.sin(2\*t))  
 return p2  
def Q2(t):  
 q2 = 2 \* abs(np.cos(t))  
 return q2  
  
# Изменения численности  
  
# Первый случай (2 регулярные армии)  
def S1(f, t):  
 s11 = -a1\*f[0] - b1\*f[1] + P1(t)  
 s12 = -c1\*f[0] - h1\*f[1] + Q1(t)  
 return s11, s12  
  
#Второй случай (1 регулярная армия против армии партизан)  
def S2(f, t):  
 s21 = -a2\*f[0] - b2\*f[1] + P2(t)  
 s22 = -c2\*f[0]\*f[1] - h2\*f[1] + Q2(t)  
 return s21, s22  
  
v = np.array([x0, y0]) # Начальные условий (Численность)  
  
# Решение  
f1 = odeint(S1, v, t)  
f2 = odeint(S2, v, t)  
  
"""  
# Первый случай (две регулярные армии)  
plt.plot(t, f1)  
plt.ylabel('Численность армии')  
plt.xlabel('Время')  
plt.legend(['Армия X (рег)', 'Армия Y (рег)'])  
  
"""  
# Второй случай (регулярная армия и партизаны)  
plt.plot(t, f2)  
plt.ylabel('Численность армии')  
plt.xlabel('Время')  
plt.legend(['Армия X (рег)', 'Армия Y (парт)'])

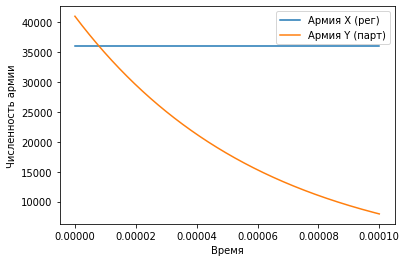
## Результат выполнения

График первого случая (рис. @fig:001)



Две регулярные армии

График второго случая (рис. @fig:002)



Регулярная армия и партизаны

# Выводы

Рассмотрел простейшую модель боевых действий – модель Ланчестера. Научился составлять системы дифференциальных уравнений изменения численностей армии и строить графики для моделей боевых действий.