

Лабораторная работа №3

Модель боевых действий

Голощапова Ирина Борисовна

25 февраля 2023

Российский университет дружбы народов, Москва, Россия

Информация

- Голощапова Ирина Борисовна
- студентка уч. группы НФИбд-01-20
- Российский университет дружбы народов
- 1032201666@pfur.ru
- <https://github.com/ibgoloshchapowa>

Вводная часть

Моделирование боевых и военных действий является важнейшей научной и практической задачей, направленной на предоставление командованию количественных оснований для принятия решений. Поэтому важно разобраться в алгоритме построения математических моделей и понять принцип их работы.

- Модель боевых действий
- Язык программирования Julia
- OpenModelica

- Разобраться в алгоритме построения математической модели.
- Рассмотреть некоторые простейшие модели боевых действий – модели Ланчестера.
- Построить графики изменения численности войск армии X и армии Y для следующих случаев:
 1. Модель боевых действий между регулярными войсками
 2. Модель ведение боевых действий с участием регулярных войск и партизанских отрядов
- Определить по графику, какая из армий одержит победу в ходе военных действий

Условие задачи. Вариант 7

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 24 000 человек, а в распоряжении страны Y армия численностью в 9 500 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:

1. Модель боевых действий между регулярными войсками

$$\frac{dx}{dt} = -0,3x(t) - 0,87y(t) + \sin(2t) + 1$$

$$\frac{dy}{dt} = -0,5x(t) - 0,41y(t) + \cos(3t) + 1$$

2. Модель ведение боевых действий с участием регулярных войск и партизанских отрядов

$$\begin{aligned}\frac{dx}{dt} &= -0,25x(t) - 0,64y(t) + \sin(2t + 4) \\ \frac{dy}{dt} &= -0,2x(t)y(t) - 0,52y(t) + \cos(t + 4)\end{aligned}$$

Рис. 2: Вариант7_2

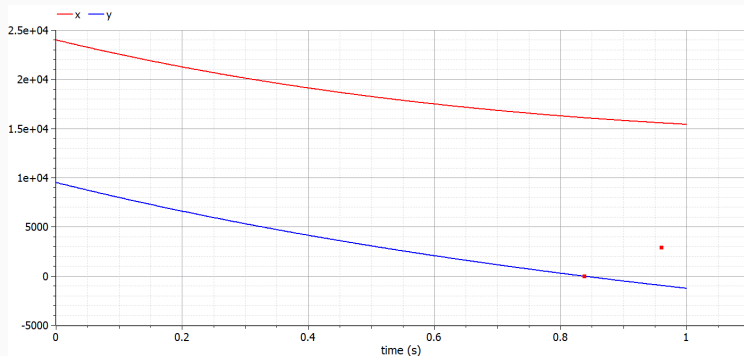
Процесс выполнения работы

1. Листинг программы в OpenModelic для случая №1

```
1 //case 1
2 model lab3
3 Real a = 0.3;
4 Real b = 0.87;
5 Real c = 0.5;
6 Real h = 0.41;
7 Real x;
8 Real y;
9 initial equation
10 x = 24000;
11 y = 9500;
12 equation
13 der(x)=-a*x-b*y+sin(2*time)+1;
14 der(y)=-c*x-h*y+cos(3*time)+1;
15 end lab3;
16 |
```

Рис. 3: Листинг программы для 1-го случая. OpenModelica

2. Модель для случая №1 в OpenModelica



В результате

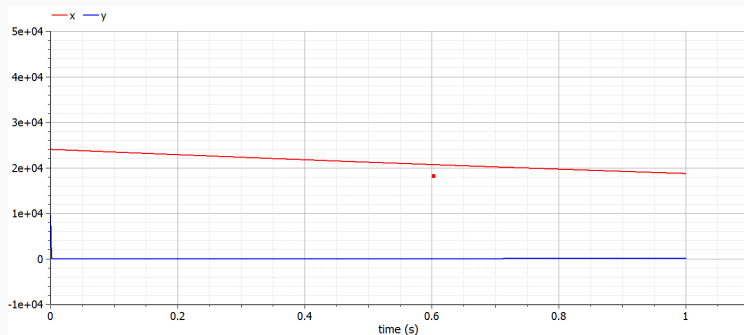
программы победу одерживает армия X.

3. Листинг программы в OpenModelic для случая №2

```
1 //case 2|
2 model lab3_2
3 Real a = 0.25;
4 Real b = 0.64;
5 Real c = 0.2;
6 Real h = 0.52;
7 Real x;
8 Real y;
9 initial equation
10 x = 24000;
11 y = 9500;
12 equation
13 der(x) = -a*x-b*y+sin(2*time+4);
14 der(y) = -c*x*y-h*y+cos(time+4);
15 end lab3_2;
```

Рис. 4: Листинг программы для 2-го случая. OpenModelica

4. Модель для случая №2 в OpenModelica



В результате программы победу одерживает армия X.

5. Листинг программы в Julia для случая №1

```
1  #case 1
2  using DifferentialEquations
3
4  function lorenz!(du, u, p, t)
5      a,b,c,h = p
6      du[1] = -a*u[1] - b*u[2] + sin(2*t) + 1
7      du[2] = -c*u[1] - h*u[2] + cos(3*t) + 1
8  end
9
10 const x = 24000
11 const y = 9500
12 u0 = [x, y]
13
14 p = (0.3, 0.87, 0.5, 0.41)
15
16 tspan = (0.0, 1.0)
17 prob = ODEProblem(lorenz!, u0, tspan, p)
18 sol = solve(prob)
19
20 using Plots; gr()
21 plot(sol)
22
23 savefig("lab3_julia_1.png")
```

Рис. 5: Листинг программы для 1-го случая. Julia

6. Модель для случая №1 в Julia

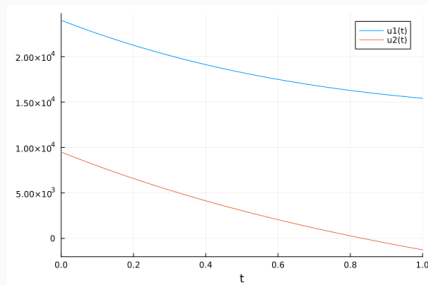


Рис. 6: Модель для 1-го случая. Julia

7. Листинг программы в Julia для случая №2

```
1  #case 2
2  using DifferentialEquations
3
4  function lorenz!(du, u, p, t)
5      a,b,c,h = p
6      du[1] = -a*u[1] - b*u[2] + sin(2*t+4)
7      du[2] = -c*u[1]*u[2] - h*u[2] + cos(t+4)
8  end
9
10 const x = 24000
11 const y = 9500
12 u0 = [x, y]
13
14 p = (0.25, 0.64, 0.2, 0.52)
15
16 tspan = (0.0, 1.0)
17 prob = ODEProblem(lorenz!, u0, tspan, p)
18 sol = solve(prob)
19
20 using Plots; gr()
21 plot(sol)
22
23 savefig("lab3_julia_2.png")
```

Рис. 7: Листинг программы для 2-го случая. Julia

8. Модель для случая №2 в Julia

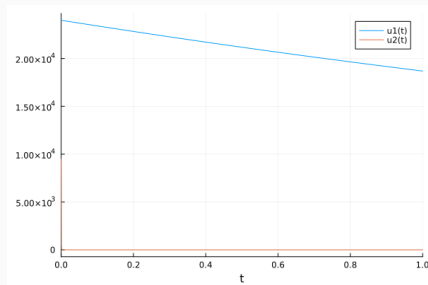


Рис. 8: Модель для 2-го случая. Julia

Результаты

- Разобралась в алгоритме построения математической модели.
- Рассмотрела модели Ланчестера.
- Построила графики изменения численности войск армии X и армии Y для следующих случаев:
 1. Модель боевых действий между регулярными войсками
 2. Модель ведение боевых действий с участием регулярных войск и партизанских отрядов
- Определила по графику, какая из армий одержит победу в ходе военных действий

В ходе лабораторной работы нам удалось построить модель боевых действий на двух языках: OpenModelica и Julia, а также с помощью построенных графиков определить, какая из двух армий одержит победу.