

Отчёт по лабораторной работе №3

Боровикова Карина Владимировна

2023, 11 февраля

RUDN University, Moscow, Russian Federation

- Важность умения работы с языками Julia и OpenModelica в части математического моделирования

- Язык Julia
- Язык OpenModelica
- Моделирование боевых действий

Цель работы:

Целью данной лабораторной работы является создание модели боевых действий с помощью языков Julia и OpenModelica. Построить соответствующие графики двух случаев ведения боевых действий.

Задачи: - Рассмотреть два случая ведения боевых действий: 1. Модель боевых действий между регулярными войсками; 2. Модель ведение боевых действий с участием регулярных войск и партизанских отрядов;

- Построить графики изменения численности войск армии X и армии Y для соответствующих случаев.

1. Рассматриваем моделирование боевых действий в двух случаях: с регулярными войсками и с участием партизанских отрядов (рис. 1).

Вариант 18

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 105 000 человек, а в распоряжении страны Y армия численностью в 95 000 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. 1: Моделирование боевых действий

2. Программируем первый случай на Julia (рис. 2).

```
1
2 using Plots
3 using DifferentialEquations
4
5 x0 = 105000
6 y0 = 95000
7
8 #= Потери, не связанные с боевыми действиями страны X=#
9 a = 0.35
10 #= Эффективность боевых действий стороны Y=#
11 b = 0.45
12 #= Эффективность боевых действий армии X=#
13 c = 0.69
14 #= Потери, не связанные с боевыми действиями страны Y=#
15 h = 0.61
16
17 #= Возможность подхода подкрепления к армии X=#
18 P(t) = 2*sin(t)
19 #= Возможность подхода подкрепления к армии Y=#
20 Q(t) = cos(t) + 1
21
22 u0 = [x0, y0]
23 p = (a, b, c, h)
24 T = [0, 5]
25
26 function F(d, u, p, t)
27     a, b, c, h = p
28     d[1] = -a*u[1] - b*u[2] + P(t)
29     d[2] = -c*u[1] - h*u[2] + Q(t)
30 end
31
32 prob = ODEProblem(F, u0, T, p)
33 sol1 = solve(prob)
```

Рис. 2: Первый случай на Julia

3. Программируем первый случай на OpenModelica (рис. 3).

```
1  model lab3 "War"
2
3  parameter Integer x0 = 105000;
4  parameter Integer y0 = 95000;
5
6  parameter Real a = 0.35;
7  parameter Real b = 0.45;
8  parameter Real c = 0.69;
9  parameter Real h = 0.61;
10
11  Real P;
12  Real Q;
13
14  Real x(start=x0);
15  Real y(start=y0);
16
17  equation
18  P = 2*sin(time);
19  Q = cos(time)+1;
20
21  der(x) = - a * x - b * y + P;
22  der(y) = - c * x - h * y + Q;
23
24  end lab3;
```

Рис. 3: Первый случай на OpenModelica

4. Программируем второй случай на Julia (рис. 4).

```
using Plots
using DifferentialEquations

x0 = 105000
y0 = 95000

#= Потери, не связанные с боевыми действиями страны X=#
a = 0.35
#= Эффективность боевых действий стороны Y=#
b = 0.73
#= Эффективность боевых действий армии X=#
c = 0.45
#= Потери, не связанные с боевыми действиями страны Y=#
h = 0.41

#= Возможность подхода подкрепления к армии X -#
P(t) = 2*sin(2*t)
#= Возможность подхода подкрепления к армии Y -#
Q(t) = cos(t) + 1

u0 = [x0, y0]
p = (a, b, c, h)
T = [0, 5]

function F(d, u, p, t)
    a, b, c, h = p
    d[1] = -a*u[1] - b*u[2] + P(t)
    d[2] = -c*u[1]*u[2] - h*u[2] + Q(t)
end

prob = ODEProblem(F, u0, T, p)
sol = solve(prob)

plt = plot!(
    sol,
    vars = [0, 1],
    color = :red,
    label = "Страна X"
```

Рис. 4: Второй случай на Julia

5. Программируем второй случай на OpenModelica (рис. 5).

```
1  model lab31 "War"
2
3  parameter Integer x0 = 105000;
4  parameter Integer y0 = 95000;
5
6  parameter Real a = 0.35;
7  parameter Real b = 0.73;
8  parameter Real c = 0.45;
9  parameter Real h = 0.41;
10
11  Real P;
12  Real Q;
13
14  Real x(start=x0);
15  Real y(start=y0);
16
17  equation
18  P = 2*sin(2*time);
19  Q = cos(time)+1;
20
21  der(x) = - a * x - b * y + P;
22  der(y) = - c * x * y - h * y + Q;
23
24  end lab31;
```

Рис. 5: Второй случай на OpenModelica

6. Запускаем код через терминал, получаем изображения для первого и второго случаев на Julia (рис. 6).

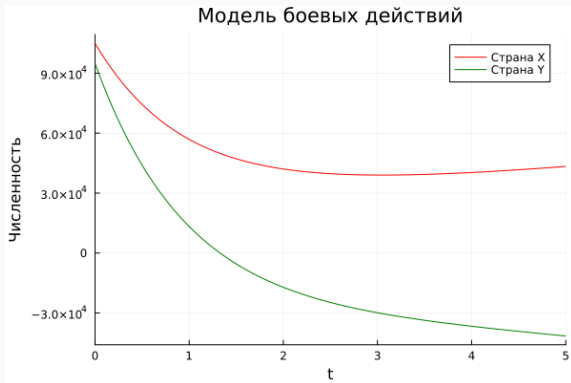


Рис. 6: Результаты выполнения кодов на Julia

7. Запускаем код через терминал, получаем изображения для первого и второго случаев на Julia (рис. 7).

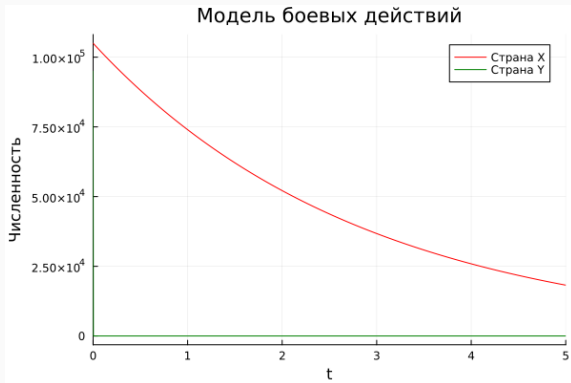


Рис. 7: Результаты выполнения кодов на Julia

8. Запускаем код через терминал, получаем изображения для первого и второго случаев на OpenModelica (рис. 8).

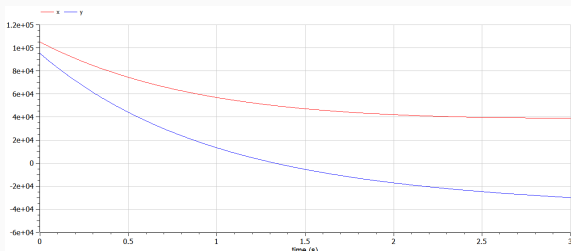


Рис. 8: Результаты выполнения кодов на Julia

9. Запускаем код через терминал, получаем изображения для первого и второго случаев на OpenModelica (рис. 9).

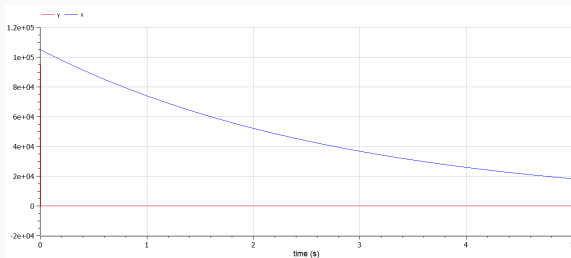


Рис. 9: Результаты выполнения кодов на Julia

Результаты

В ходе выполнения данной лабораторной работы мы создали модели боевых действий с помощью языков Julia и OpenModelica, а также построили соответствующие графики двух случаев ведения боевых действий.