

# **Отчёт по лабораторной работе №3**

**Математическое моделирование**

Лушин Артём Андреевич

# Содержание

1	Цель работы	4
2	Выполнение лабораторной работы	5
3	Вывод	10

## Список иллюстраций

2.1	Пример 1: код Julia . . . . .	5
2.2	Пример 1: график Julia . . . . .	5
2.3	Пример 1: код OpenModelica . . . . .	6
2.4	Пример 1: график OpenModelica . . . . .	7
2.5	Пример 2: код Julia . . . . .	7
2.6	Пример 2: график Julia . . . . .	8
2.7	Пример 2: код OpenModelica . . . . .	8
2.8	Пример 2: графика OpenModelica . . . . .	9

# 1 Цель работы

Реализовать модель боевых действий на языках OpenModelica и Julia.

## 2 Выполнение лабораторной работы

- 1) Я написал программу для построения графика модели боевых действий между регулярными войсками на языке Julia. В результате мы можем увидеть, что выигрывает армия У.

```
[13]: using DifferentialEquations, Plots;

function reg(u, p, t)
    x, y = u
    a, b, c, h = p
    dx = -a*x - b*y*sin(t+1)
    dy = -c*x - h*y*cos(t+2)
    return [dx, dy]
end

u0 = [32500, 13800]
p = [0.12, 0.54, 0.4, 0.27]
tspan = (0,1)
prob = ODEProblem(reg, u0, tspan, p)
sol = solve(prob, Tsit5())
plot(sol, title = "Модель боевых действий между регулярными войсками", label = ["Армия x" "Армия y"], xaxis = "Время", yaxis = "Численность армий")
```

Рис. 2.1: Пример 1: код Julia

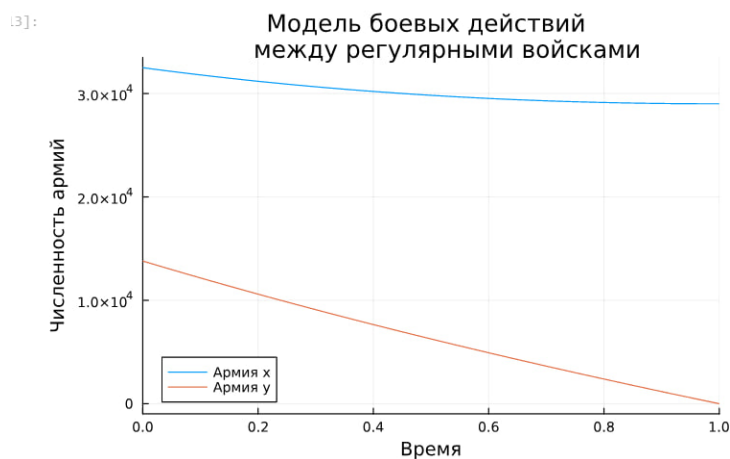


Рис. 2.2: Пример 1: график Julia

- 2) Написал такую же программу, но на языке OpenModelica. На этом языке также построил график и сравнил графики на двух языках. Графики идентичные.

```
model mat
  parameter Real a=0.12;
  parameter Real b=0.54;
  parameter Real c=0.4;
  parameter Real h=0.27;

  parameter Real x0=32500;
  parameter Real y0=13800;

  Real x(start=x0);
  Real y(start=y0);

equation

  der(x)=-a*x-b*y+sin(time+1);
  der(y)=-c*x-h*y+cos(time+2);

end mat;
```

Рис. 2.3: Пример 1: код OpenModelica

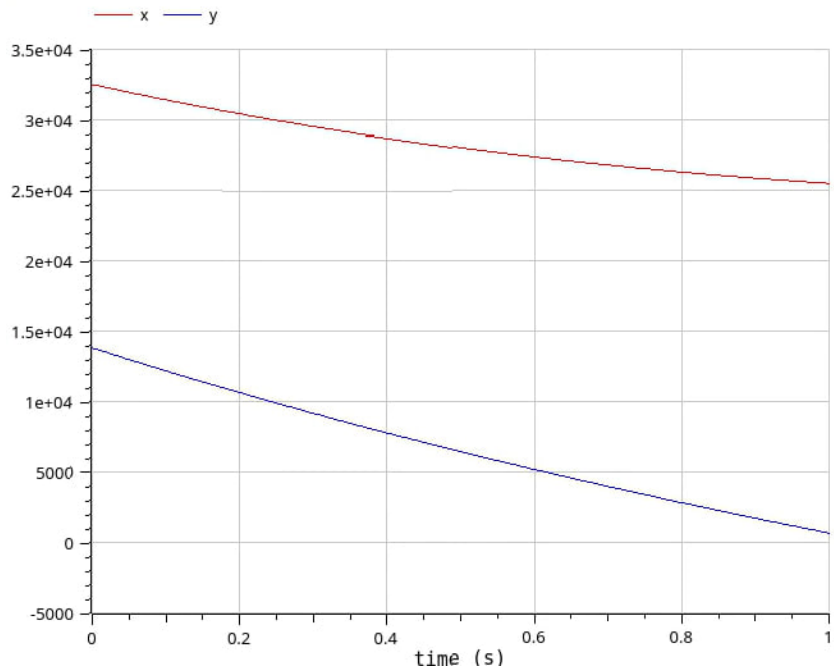


Рис. 2.4: Пример 1: график OpenModelica

- 3) Написал программу для построения графика модели боевых действий регулярных армий с участием партизанских отрядов на языке Julia. В данном случае побеждает армия X, а численность армии Y практически сразу уходит в ноль.

```
[15]: function reg_part(u,p,t)
      x,y = u
      a,b,c,h = p
      dx = -a*x - b*y + c*sin(2*pi*t)
      dy = -c*x + y - h*y + 2*cos(t)
      return [dx,dy]
    end

    u0 = [32500, 13000]
    p = [0.25, 0.8, 0.62, 0.13]
    tspan = (0,1)
    prob2 = ODEProblem(reg_part, u0, tspan, p)
    sol2 = solve(prob2, Tsit5())

    plot(sol2, title = "Модель боевых действий с участием регулярных войск и партизанских отрядов", label = ["Армия X" "Армия Y"], xaxis = "Время", yaxis = "
```

Рис. 2.5: Пример 2: код Julia

[15]: Левых действий с участием регулярных войск и парти:

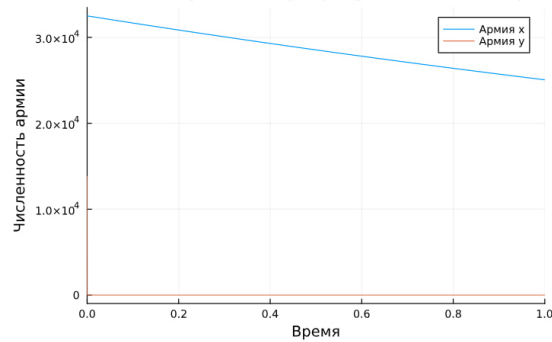


Рис. 2.6: Пример 2: график Julia

- 4) Написал аналогичную программу на языке OpenModelica. Так же как и в случае с Julia, численность армии Y практически сразу уходит в ноль. На данном графике это лучше видно, так как больше интервал.

```

1  model mar3_2
2
3      parameter Real a=0.26;
4      parameter Real b=0.8;
5      parameter Real c=0.62;
6      parameter Real h=0.13;
7
8      parameter Real x0 = 32500;
9      parameter Real y0 = 13800;
10
11     Real x(start=x0);
12     Real y(start=y0);
13
14     equation
15
16     der(x)=-a*x-b*y+sin(2*time);
17     der(y)=-c*x*y-h*y+cos(time);
18
19 end mar3_2;
```

Рис. 2.7: Пример 2: код OpenModelica



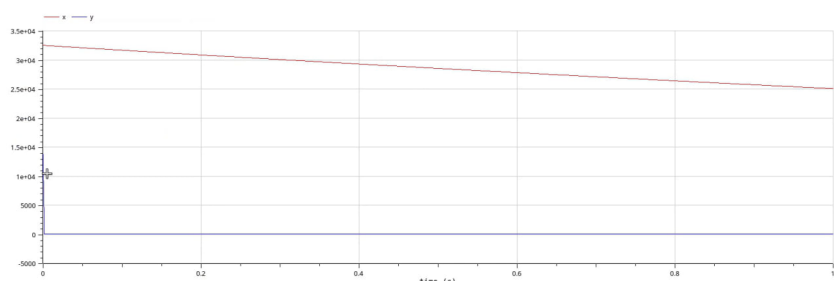


Рис. 2.8: Пример 2: графика OpenModelica

## 3 Вывод

Я реализовал модель боевых действий на языке Julia и OpenModelica.