

Лабораторная работа № 3

Тарусов Артём Сергеевич

2023, Москва

Целью данной работы является построение модели боевых действий.

- Построение модели боевых действий между регулярными войсками на языках Julia и OpenModelica
- Построение модели боевых действий с участием регулярных войск и партизанских отрядов на языках Julia и OpenModelica

Опишем начальные значения согласно варианту 8 на языке Julia.

```
x0 ::Int64 = 19300  
y0 ::Int64 = 39000
```

Рис. 1: Начальные значения

Укажем коэффициенты a , b , c , d .

```
a ::Float64 = 0.46 # константа, характеризующая степень влияния различных факторов на потери
b ::Float64 = 0.7  # эффективность боевых действий армии y
c ::Float64 = 0.82 # эффективность боевых действий армии x
h ::Float64 = 0.5  # константа, характеризующая степень влияния различных факторов на потери
```

Рис. 2: Коэффициенты a , b , c , d для первого случая на языке Julia

Опишем функцию P подхода подкрепления армии X и функцию Q для армии Y . Составим систему дифференциальных уравнений, описывающую противостояние регулярных войск.

```
function P(t)
    p = sin(0.5t)
end

function Q(t)
    q = cos(1.5t)
end

function ode_fn(du, u, p, t)
    x, y = u
    du[1] = - a*u[1] - b*u[2] + P(t) # изменение численности первой армии
    du[2] = - c*u[1] - h*u[2] + Q(t) # изменение численности второй
end
```

Получим решение системы дифференциальных уравнений.

```
v0 ::Array{Any} = [x0, y0]
tspan = (0.0, 1.0)
prob = ODEProblem(ode_fn, v0, tspan)
sol = solve(prob, dtmax=0.1)

X = [u[1] for u in sol.u]
Y = [u[2] for u in sol.u]
T = [t for t in sol.t]
```

Рис. 4: Решение системы дифференциальных уравнений для первого случая на языке Julia

Построим графики по полученным данным.

```
plt = plot(  
    dpi=300,  
    title="Модель боевых действий",  
    legend=true)  
  
plot!(  
    plt,  
    T,  
    X,  
    label="Численность армии x",  
    color=:blue)  
  
plot!(  
    plt,  
    T,  
    Y,  
    label="Численность армии y",  
    color=:red)  
  
savefig(plt, "out/lab03_1.png")
```

Рис. 5: Построение графика для первого случая на языке Julia

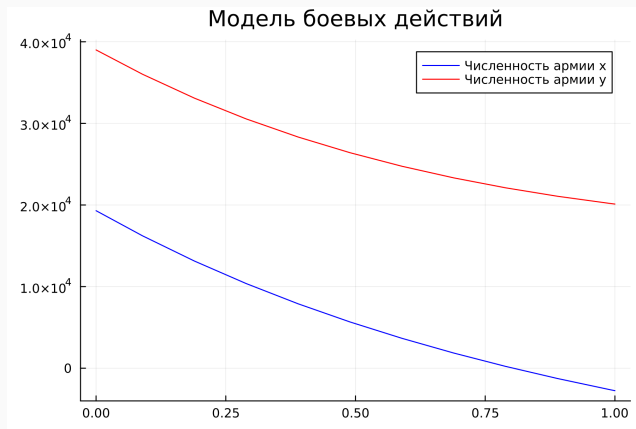


Рис. 6: Модель противостояния регулярных войск, построенная на языке Julia

Для второго случая начальные значения численности армий X и Y остаются прежними. Определим коэффициенты a , b , c , d

```
a ::Float64 = 0.38 # константа, характеризующая степень влияния различных факторов на потери  
b ::Float64 = 0.73 # эффективность боевых действий армии y  
c ::Float64 = 0.5 # эффективность боевых действий армии x  
h ::Float64 = 0.28 # константа, характеризующая степень влияния различных факторов на потери
```

Рис. 7: Коэффициенты a , b , c , d для второго случая на языке Julia

Опишем функции P , Q и систему дифференциальных уравнений для второго случая

```
function P(t)
    p = sin(2t) + 1
end

function Q(t)
    q = cos(2t)
end

function ode_fn(du, u, p, t)
    x, y = u
    du[1] = - a*u[1] - b*u[2] + P(t) # изменение численности первой армии
    du[2] = - c*u[1]*u[2] - h*u[2] + Q(t) # изменение численности второй
end
```

Рис. 8: Составление функций для моделирования боевых действий с участием регулярных войск и партизанских отрядов на языке Julia

По аналогии с первым случаем построим модель для второго

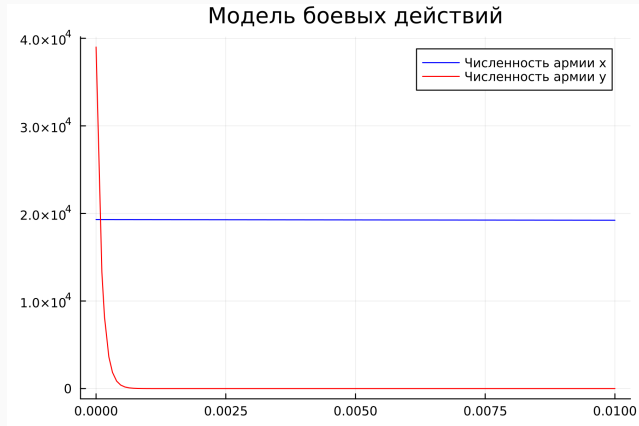


Рис. 9: Модель противостояния регулярных войск с участием партизанских отрядов, построенная на языке Julia

Построим программу, моделирующую боевые действия с участием регулярных войск с помощью OpenModelica

```
1  model Lab03_1
2  Real x;
3  Real y;
4  Real a = 0.46;
5  Real b = 0.7;
6  Real c = 0.82;
7  Real d = 0.5;
8  Real t = time;
9  initial equation
10 x = 19300;
11 y = 39000;
12 equation
13 der(x) = -a*x - b*y + sin(0.5*t);
14 der(y) = -c*x - d*y + cos(1.5*t);
15 end Lab03_1;
```

Рис. 10: Построение модели для первого случая на языке OpenModelica

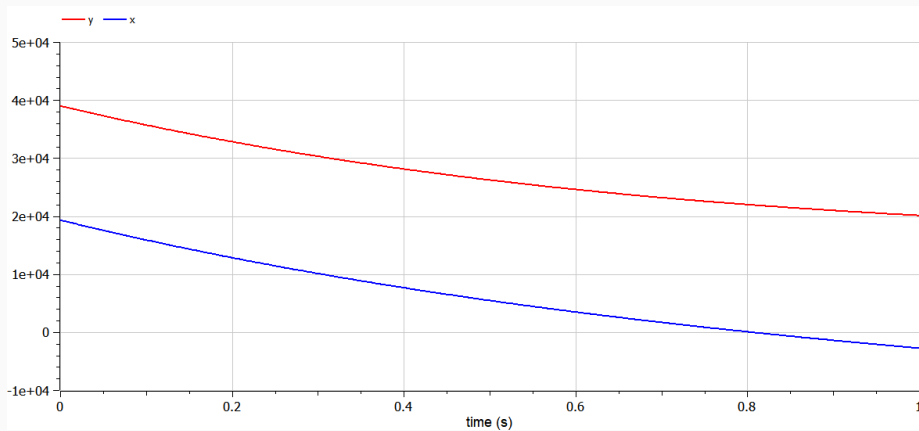


Рис. 11: Модель противостояния регулярных войск, построенная на языке OpenModelica

Построим программу, моделирующую боевые действия с участием регулярных войск и партизанских отрядов с помощью OpenModelica.

```
1  model Lab03_2
2  Real x;
3  Real y;
4  Real a = 0.38;
5  Real b = 0.73;
6  Real c = 0.5;
7  Real d = 0.28;
8  Real t = time;
9  initial equation
10 x = 19300;
11 y = 39000;
12 equation
13 der(x) = -a*x - b*y + sin(2*t) + 1;
14 der(y) = -c*x*y - d*y + cos(2*t);
15 end Lab03_2;
```

Рис. 12: Построение модели для второго случая на языке OpenModelica

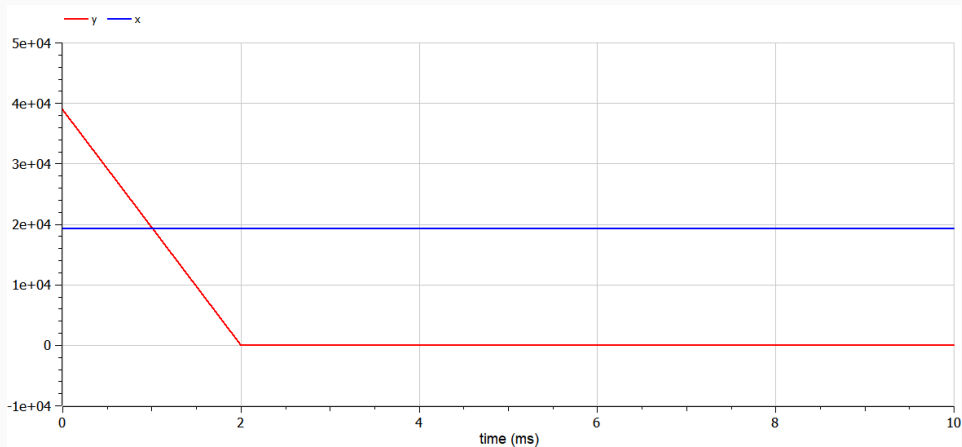


Рис. 13: Модель противостояния регулярных войск с участием партизанских отрядов, построенная на языке OpenModelica

В итоге проделанной работы мы построили по две модели на языках Julia и OpenModelica. В ходе проделанной работы можем сделать вывод, что OpenModelica лучше приспособлен для моделирование процессов, протекающих во времени. Построение моделей боевых действий на языке openModelica занимает меньше строк, чем аналогичное построение на Julia.