

Лабораторная работа №5

Модель Лотки-Вольтерры

Абу Сувейлим Мухаммед Мунифович

Содержание

1	Цель работы	4
2	Задание	5
3	Теоретическое введение	6
4	Выполнение лабораторной работы	7
4.1	Моделирование на языке программирования Julia	7
4.1.1	Построение графика зависимости численности хищников от численности жертв	7
4.2	Моделирование на языке программирования OpenModelica . . .	11
4.3	Исходный код	13
4.3.1	Julia	13
4.3.2	OpenModelica	18
5	Вывод	21
6	Библиография	22

Список иллюстраций

4.1	График зависимости численности хищников от численности жертв	9
4.2	График изменения численности хищников и численности жертв при заданных начальных условиях	10
4.3	Стационарное состояние системы	11
4.4	График зависимости численности хищников от численности жертв в OpenModelica	12
4.5	График изменения численности хищников и численности жертв при заданый начальных условиях в OpenModelica	13
4.6	стационарное состояние в OpenModelica	13

1 Цель работы

- Целью работы является познакомиться с моделью хищник-жертва.

2 Задание

1. Постройте график зависимости численности хищников от численности жертв;
2. графики изменения численности хищников и численности жертв при следующих начальных условиях: $x_0 = 8, y_0 = 16$;
3. найти стационарное состояние системы.

3 Теоретическое введение

Модель Лотки — Вольтерры (модель Лотки — Вольтерра) [1] — модель взаимодействия двух видов типа «хищник — жертва», названная в честь своих авторов (Лотка, 1925; Вольтерра 1926), которые предложили модельные уравнения независимо друг от друга.

Такие уравнения можно использовать для моделирования систем «хищник — жертва», «паразит — хозяин», конкуренции и других видов взаимодействия между двумя видами [2].

В математической форме предложенная система имеет следующий вид:

$$\begin{aligned}\frac{dx}{dt} &= ax(t) - bx(t)y(t) \\ \frac{dy}{dt} &= -cx(t) + dx(t)y(t)\end{aligned}$$

В этой модели x — число жертв, y — число хищников. Коэффициент a описывает скорость естественного прироста числа жертв в отсутствие хищников, с естественное вымирание хищников, лишенных пищи в виде жертв. Вероятность взаимодействия жертвы и хищника считается пропорциональной как количеству жертв, так и числу самих хищников (xy). Каждый акт взаимодействия уменьшает популяцию жертв, но способствует увеличению популяции хищников (члены $-bxy$ и dxy в правой части уравнения).

4 Выполнение лабораторной работы

4.1 Моделирование на языке программирования Julia

4.1.1 Построение графика зависимости численности хищников от численности жертв

1. Во-первых, я использовал пакеты Plots и DifferentialEquations для построения графиков и для решения дифференциальных уравнений, соответственно.

```
using Plots
```

```
using DifferentialEquations
```

2. Инициализировал нужны нам константы и функции в модели. $a = 0.83$ - это коэффициент естественной смертности хищников; $b = 0.083$ - это коэффициент естественного прироста жертв $c = 0.82$ - это коэффициент увеличения числа хищников $d = 0.082$ - это коэффициент смертности жертв $x_0 = 8$ - это начальное число жертв $y_0 = 16$ - это начальное число хищников

```
#начальные значения
```

```
a= 0.83 #коэффициент естественной смертности хищников
```

```
b= 0.083#коэффициент естественного прироста жертв
```

```
c= 0.82 #коэффициент увеличения числа хищников
```

```

d= 0.082 #коэффициент смертности жертв
x0 = 8 #начальное число жертв
y0 = 16 #начальное число хищников
#система дифф уравнений 1-го порядка для модели Лотки-Вольтерры

```

```

function modelLotkaVolterra(du, u, p, t)
    x, y = u
    du[1] = a*u[1] -b*u[1]*u[2]
    du[2] = -c*u[2] + d*u[1]*u[2]
end

```

3. Далее я обозначал интервал времени.

```

tspan = (0, 100) #произвольный интервал времени

```

4. Здесь я дал аргументы для функции ODEProblem которая указывает на дифф уравнение. Далее, я уравнение решил. Шаг времени = 0.05

```

prob = ODEProblem(modelLotkaVolterra, u0, tspan)
sol = solve(prob, dtmax = 0.05)

```

5. Здесь я переименовал названия переменных.

```

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

```

6. Далее я подготовил пространство для первого графика.

```

pltOne = plot(dpi = 300, legend = false)

```

7. Наконец, я построил график зависимости численности хищников от численности жертв.

```

plot!(
    pltOne,

```



```

X,
Y,
title = "График зависимости численности хищников от численности жертв",
titlefont = font(12,"Computer Modern"),
xlabel = "X - численность жертв",
ylabel = "Y - численность хищников",
xtickfontsize=8,
xguidefontsize=12,
xguidefont = "Computer Modern",
ytickfontsize=8,
yguidefontsize=12,
yguidefont = "Computer Modern",
color=:red)

```

8. Получуный график.

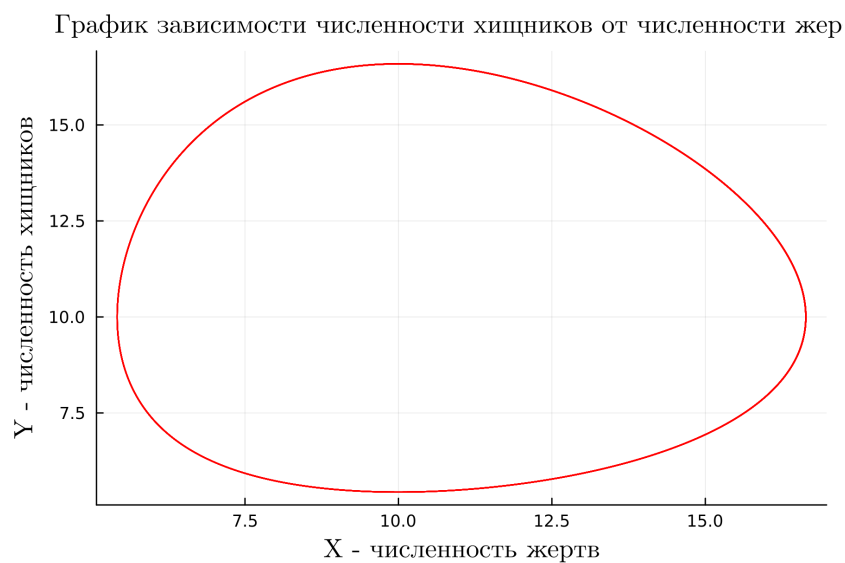


Рис. 4.1: График зависимости численности хищников от численности жертв

9. Далее я подготовил пространство для второго графика.

```

pltTwo = plot(dpi = 300, legend = false)

```

11. Я построил график изменения численности хищников и численности жертв при заданных начальных условиях.

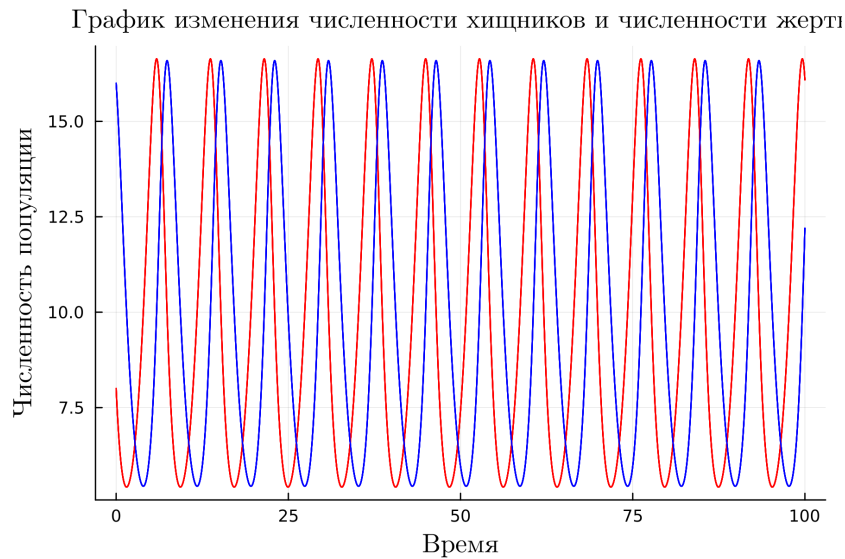


Рис. 4.2: График изменения численности хищников и численности жертв при заданных начальных условиях

12. Для нахождения стационарного состояния системы нужно чтобы начальное точка была в координатах $x_0 = \frac{c}{d}, y_0 = \frac{a}{b}$.

#начальные значения

a= 0.83 #коэффициент естественной смертности хищников

b= 0.083#коэффициент естественного прироста жертв

c= 0.82 #коэффициент увеличения числа хищников

d= 0.082 #коэффициент смертности жертв

x0 = c / d #начальное число жертв

y0 = a / b#начальное число хищников

13. Все остальное как и было.
14. Получаем следующий график, где у нас численность популяций не меняется никогда. Также получилось что у нас численность популяций хищников и жертв одинаковые.

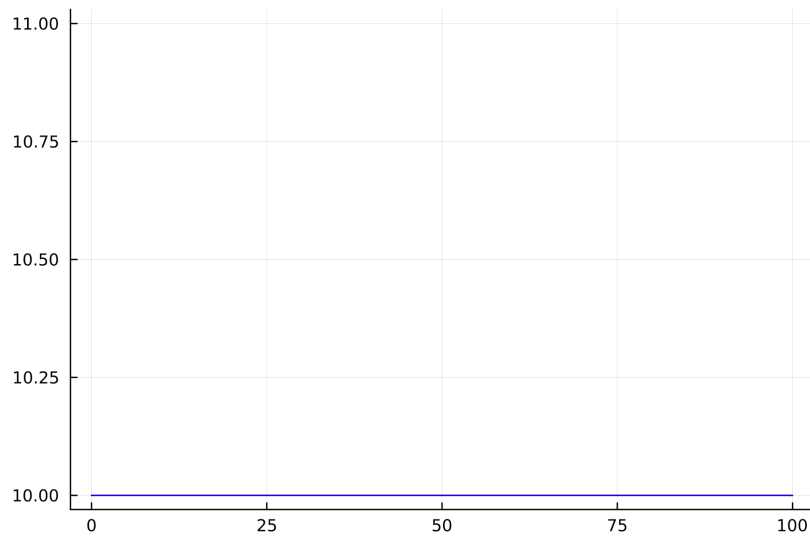


Рис. 4.3: Стационарное состояние системы

4.2 Моделирование на языке программирования

OpenModelica

1. В OpenModelica все проще. Я просто переписал код из Julia. В этой программе все величины имеют тот же смысл, что и в Julia. Переменная t указывает на время.

```
model lab5_1
```

```
Real x;
```

```
Real y;
```

```
Real a = 0.83;
```

```
Real b = 0.083;
```

```
Real c = 0.82;
```

```
Real d = 0.082;
```

```
Real t = time;
```

```
initial equation
```

```
x = 8;
```

```
y = 16;
```

```
equation
```

```
der(x) = -a*x + b*x*y;
```

```
der(y) = c*y-b*x*y;
```

```
end lab5_1;
```

2. График зависимости численности хищников от численности жертв в OpenModelica.

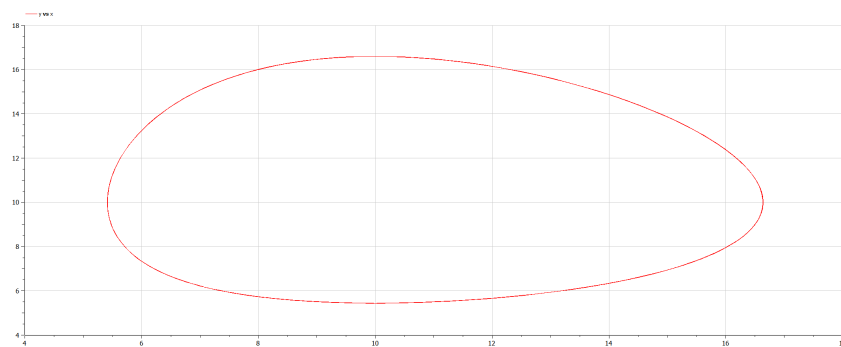


Рис. 4.4: График зависимости численности хищников от численности жертв в OpenModelica

3. График изменения численности хищников и численности жертв при заданных начальных условиях в OpenModelica

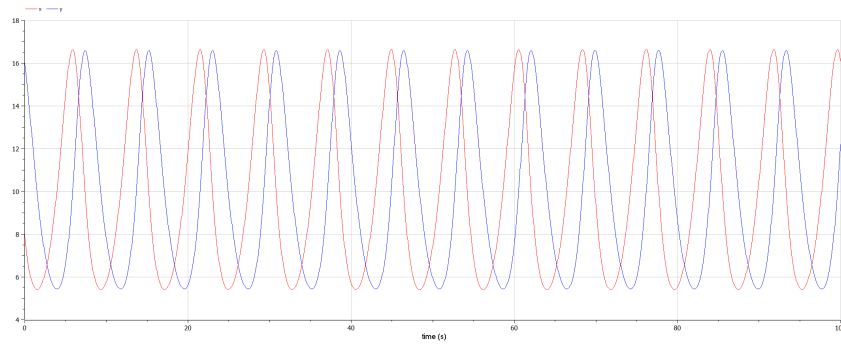


Рис. 4.5: График изменения численности хищников и численности жертв при заданый начальных условиях в OpenModelica

4. Как и в Julia, для нахождения стационарное состояние системы нужно чтобы начадьное точка была в координатах $x_0 = \frac{c}{d}$, $y_0 = \frac{a}{b}$.

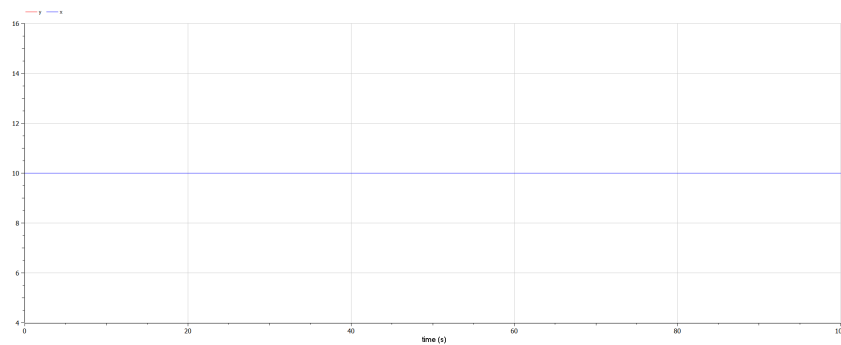


Рис. 4.6: стационарное состояние в OpenModelica

4.3 Исходный код

4.3.1 Julia

1. Код в случае график зависимости численности хищников от численности жертв, а также графики изменения численности хищников и численности жертв при заданных начальных условиях на Julia

```
using Plots
using DifferentialEquations
```

```

#Вариант 36
1032215135%70 + 1

#начальные значения

a= 0.83 #коэффициент естественной смертности хищников
b= 0.083#коэффициент естественного прироста жертв
c= 0.82 #коэффициент увеличения числа хищников
d= 0.082 #коэффициент смертности жертв
x0 = 8 #начальное число жертв
y0 = 16 #начальное число хищников

#система дифф уравнений 1-го порядка для модели Лотки-Вольтерры

function modelLotkaVolterra(du, u, p, t)
    x, y = u
    du[1] = a*u[1] -b*u[1]*u[2]
    du[2] = -c*u[2] + d*u[1]*u[2]
end

u0 = [x0, y0]

tspan = (0, 100) #произвольный интервал времени

prob = ODEProblem(modelLotkaVolterra, u0, tspan)

sol = solve(prob, dtmax = 0.05)

```

```

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

pltOne = plot(dpi = 300, legend = false)

plot!(
    pltOne,
    X,
    Y,
    title = "График зависимости численности хищников от численности жертв",
    titlefont = font(12,"Computer Modern"),
    xlabel = "X - численность жертв",
    ylabel = "Y - численность хищников",
    xtickfontsize=8,
    xguidefontsize=12,
    xguidefont = "Computer Modern",
    ytickfontsize=8,
    yguidefontsize=12,
    yguidefont = "Computer Modern",
    color=:red)

savefig(pltOne, "lab5_1.png")

pltTwo = plot(dpi = 300, legend = false, label = ["Численность жертв" "Чис

plot!(
    pltTwo,
    Time,

```

```

X,
label = ["Численность жертв"],
title = "График изменения численности хищников и численности жертв",
titlefont = font(8,"Computer Modern"),
xlabel = "Время",
ylabel = "Численность популяции",
xtickfontsize=8,
xguidefontsize=12,
xguidefont = "Computer Modern",
ytickfontsize=8,
yguidefontsize=12,
yguidefont = "Computer Modern",
color=:red)

plot!(
    pltTwo,
    Time,
    Y,
    label = ["Численность хищников"],
    title = "График изменения численности хищников и численности жертв",
    titlefont = font(12,"Computer Modern"),
    xlabel = "Время",
    ylabel = "Численность популяции",
    xtickfontsize=8,
    xguidefontsize=12,
    xguidefont = "Computer Modern",
    ytickfontsize=8,
    yguidefontsize=12,
    yguidefont = "Computer Modern",
    color=:blue)

```


2. Код стационарное состояние системы на Julia.

```
using Plots
using DifferentialEquations

#начальные значения

a= 0.83 #коэффициент естественной смертности хищников
b= 0.083#коэффициент естественного прироста жертв
c= 0.82 #коэффициент увеличения числа хищников
d= 0.082 #коэффициент смертности жертв
x0 = c / d #начальное число жертв
y0 = a / b#начальное число хищников

#система дифф уравнений 1-го порядка для модели Лотки-Вольтерры

function stationaryModelLotkaVolterra(du, u, p, t)
    x, y = u
    du[1] = a*u[1] -b*u[1]*u[2]
    du[2] = -c*u[2] + d*u[1]*u[2]
end

u0 = [x0, y0]

tspan = (0, 100) #произвольный интервал времени

prob = ODEProblem(stationaryModelLotkaVolterra, u0, tspan)

sol = solve(prob, dtmax = 0.05)
```

```

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

pltTwo = plot(dpi = 300, legend = false)

plot!(
    pltTwo,
    Time,
    X,
    label = "Численность жертв",
    color=:red)
plot!(
    pltTwo,
    Time,
    Y,
    label = "Численность хищников",
    color=:blue)

savefig(pltTwo, "lab5_2_2")

```

4.3.2 OpenModelica

1. Код в случае график зависимости численности хищников от численности жертв, а также графики изменения численности хищников и численности жертв при заданных начальных условиях в OpenModelica

```
model lab5_1
```

```
Real x;
```

```

Real y;
Real a = 0.83;
Real b = 0.083;
Real c = 0.82;
Real d = 0.082;

Real t = time;

initial equation

x = 8;
y = 16;

equation

der(x) = -a*x + b*x*y;
der(y) = c*y-b*x*y;

end lab5_1;

```

2. Код стационарное состояние системы в OpenModelica

```

model lab5_2

```

```

Real x;
Real y;
Real a = 0.83;
Real b = 0.083;
Real c = 0.82;
Real d = 0.082;

```

```
initial equation
```

```
x = c / d;
```

```
y = a / b;
```

```
equation
```

```
der(x) = a*x - b*x*y;
```

```
der(y) = -c*y+d*x*y;
```

```
end lab5_2;
```

5 Вывод

- Жесткую модель всегда надлежит исследовать на структурную устойчивость полученных при ее изучении результатов по отношению к малым изменениям модели (делаящим ее мягкой).
- Если начальные значения задать в стационарном состоянии $x(0) = x_0$ и $y(0) = y_0$, то в любой момент времени численность популяций изменяться не будет

6 Библиография

1. Турчин П.В. Лекция №14. Популяционная динамика. Биологическое образование в МФТИ, 2012.
2. Ю. О. Основ экологии. Мир, 1986. 376 с.