

Лабораторная работа №5. Модель хищник-жертва

Выполнила: Лебедева Ольга Андреевна

Преподаватель Кулябов Дмитрий Сергеевич д.ф.-м.н., профессор кафедры теории вероятностей и кибербезопасности

2024

Российский университет дружбы народов, Москва, Россия

Построить простейшую модель взаимодействия двух видов типа «хищник — жертва» - модель Лотки-Вольтерры.

Модель Лотки-Вольтерры является классическим примером математической модели взаимодействия между хищниками и их жертвами в экологии. Названная в честь двух итальянских ученых, Альфредо Лотки и Витторио Вольтерры, она представляет собой систему дифференциальных уравнений, которые описывают динамику популяций обоих видов во времени.

В модели учитывается два вида организмов: хищники и их жертвы. Предполагается, что популяции обоих видов развиваются в изолированной среде без внешних влияний [1].

Вариант 17

Для модели «хищник-жертва»:

$$\begin{cases} \frac{dx}{dt} = -0.69x(t) + 0.068x(t)y(t) \\ \frac{dy}{dt} = 0.67y(t) - 0.066x(t)y(t) \end{cases}$$

Постройте график зависимости численности хищников от численности жертв, а также графики изменения численности хищников и численности жертв при следующих начальных условиях $x_0 = 4$, $y_0 = 11$. Найдите стационарное состояние системы.

Выполнение лабораторной работы

Напишем код на Julia для случая 1: нестационарное состояние системы.

```
using Plots, DifferentialEquations

x0 = 4
y0 = 11

a = 0.69
b = 0.068
c = 0.67
d = 0.66

function ode_fn(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
```

```
v0 = [x0, y0]
tspan = (0.0, 60.0)
prob = ODEProblem(ode_fn, v0, tspan)
sol = solve(prob, dtmax=0.05)
X = [u[1] for u in sol.u]
Y = [u[2] for u in sol.u]
T = [t for t in sol.t]
```

```
plt = plot(dpi=300, legend=false)
plot!(plt, X, Y, color=:blue)
savefig(plt, "lab05_1.png")
```

```
plt2 = plot(dpi=300, legend=true)
plot!(plt2, T, X, label="Численность жертв", color=:red)
plot!(plt2, T, Y, label="Численность хищников", color=:gr)
savefig(plt2, "lab05_2.png")
```

Запустим код при помощи командной строки и получим два изображения: См. рис. 1, См. рис. 2

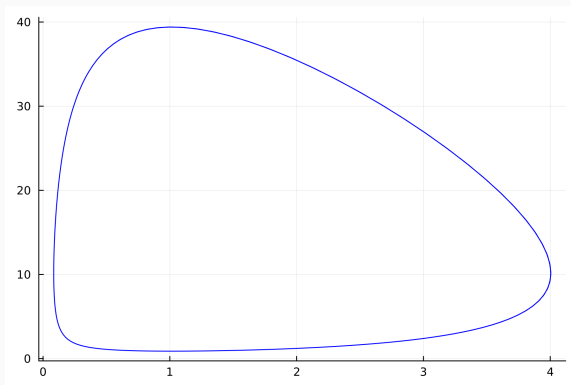


Рис. 1: Динамика популяций хищников относительно жертв

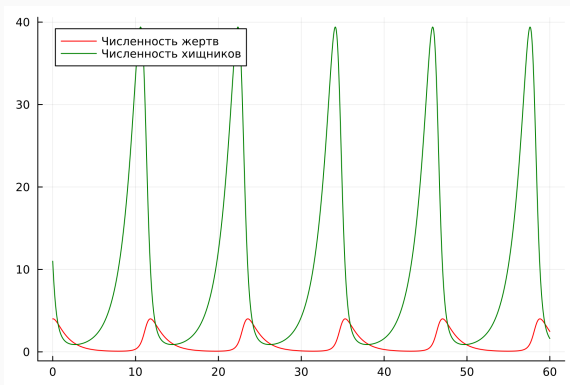


Рис. 2: Изменение популяций хищников и жертв по времени

Напишем код на Julia для случая 2: стационарное состояние системы.

```
using Plots, DifferentialEquations

a = 0.69
b = 0.068
c = 0.67
d = 0.66

x0 = c / d
y0 = a / b

function ode_fn(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
```

```
v0 = [x0, y0]
tspan = (0.0, 60.0)
prob = ODEProblem(ode_fn, v0, tspan)
sol = solve(prob, dtmax=0.05)
X = [u[1] for u in sol.u]
Y = [u[2] for u in sol.u]
T = [t for t in sol.t]

plt2 = plot(dpi=300, legend=true)
plot!(plt2, T, X, label="Численность жертв", color=:red)
plot!(plt2, T, Y, label="Численность хищников", color=:gr
savefig(plt2, "lab05_3.png")
```

Запустим код при помощи командной строки и получим изображение: См. рис. 3

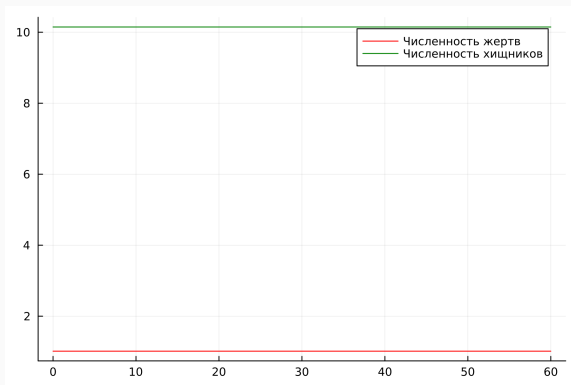


Рис. 3: Стационарное состояние системы

В стационарном состоянии решение будет представлять собой точку.

Напишем код на OpenModelica для случая 1: нестационарное состояние системы.

```
model PredatorPrey
parameter Real a = 0.69;
parameter Real b = 0.068;
parameter Real c = 0.67;
parameter Real d = 0.66;

parameter Real x0 = 4;
parameter Real y0 = 11;

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

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

end PredatorPrey;
```

Запустим код при помощи кнопок “проверить модель” -> “установки симуляции” -> “симулировать”. Не забываем в настройках указать заданные нам начальные условия См. рис. 4, См. рис. 5

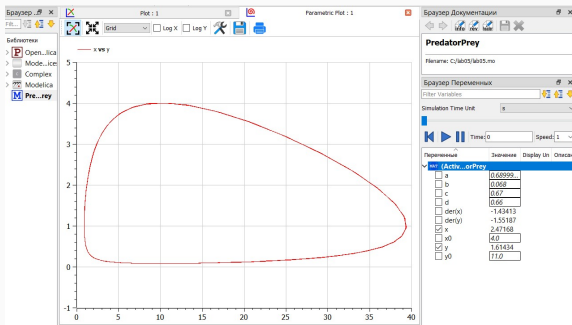


Рис. 4: Динамика популяций хищников относительно жертв

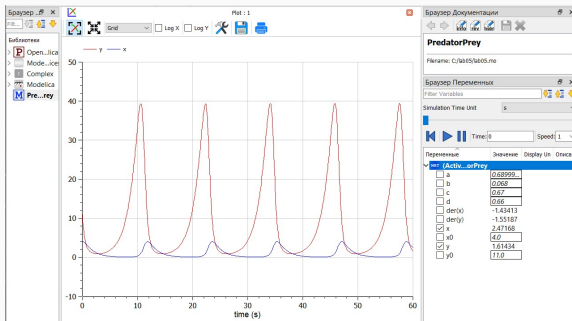


Рис. 5: Изменение популяций хищников и жертв по времени

Напишем код для случая 2: стационарное состояние системы.

```
model lab05_2
  Real a = 0.69;
  Real b = 0.068;
  Real c = 0.67;
  Real d = 0.066;
  Real x;
  Real y;
  initial equation
    x = c / d;
    y = a / b;
  equation
    der(x) = -a*x + b*x*y;
    der(y) = c*y - d*x*y;
end lab05_2;
```


Запустим код. Нажимаем галочки x и y для отображения графиков: См. рис. 6

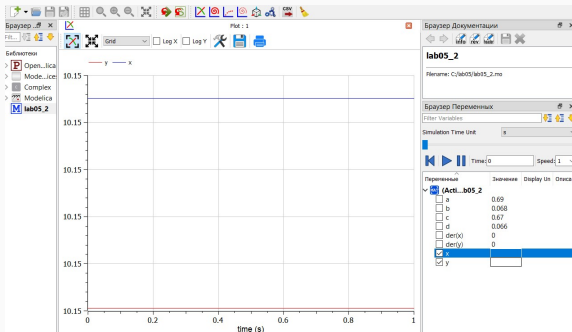


Рис. 6: Стационарное состояние системы

Построили простейшую модель взаимодействия двух видов типа «хищник — жертва» - модель Лотки-Вольтерры.

[1] Модель хищник-жертва:

https://math-it.petrso.ru/users/semenova/MathECO/Lectures/Lotka_Volterra.pdf