Лабораторная работа №5

Модель хищник-жертва

Го Чаопен

Содержание

Список иллюстраций

1 Цель работы

Рассмотреть модель хищник-жертва. Построить вышеуказанную модель средствами OpenModellica и Julia.

2 Задачи

Для модели «хищник-жертва»:

$$\begin{cases} \frac{dx}{dt} = -0.71x(t) + 0.046x(t)y(t) \\ \frac{dy}{dt} = 0.64y(t) - 0.017x(t)y(t) \end{cases}$$

1. Построить график зависимости численности хищников от численности жертв, а также графики изменения численности хищников и численности жертв при следующих начальных условиях: $x_0 = 4, y_0 = 12$.

2. Найти стационарное состояние системы.

3 Среда

- Julia — высокоуровневый высокопроизводительный свободный язык программирования с динамической типизацией, созданный для математических вычислений. [1]
- OpenModelica — свободное открытое программное обеспечение для моделирования, симуляции, оптимизации и анализа сложных динамических систем. [2]

4 Теоретическое введение

Простейшая модель взаимодействия двух видов типа «хищник — жертва» - модель Лотки-Вольтерры. Данная двувидовая модель основывается на следующих предположениях [3]:

1. Численность популяции жертв x и хищников y зависят только от времени (модель не учитывает пространственное распределение популяции на занимаемой территории);
2. В отсутствии взаимодействия численность видов изменяется по модели Мальтуса, при этом число жертв увеличивается, а число хищников падает;
3. Естественная смертность жертвы и естественная рождаемость хищника считаются несущественными;
4. Эффект насыщения численности обеих популяций не учитывается;
5. Скорость роста численности жертв уменьшается пропорционально численности хищников;

$$\begin{cases} \frac{dx}{dt} = ax(t) - bx(t)y(t) \\ \frac{dy}{dt} = -cy(t) + dx(t)y(t) \end{cases}$$

В этой модели x – число жертв, y - число хищников. Коэффициент a описывает скорость естественного прироста числа жертв в отсутствие хищников, - 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).

Стационарное состояние данной системы (положение равновесия, не зависящее от времени решение) будет в точке:

$$x_0 = \frac{c}{d},\ y_0 = \frac{a}{b}$$

Если начальные значения задать в стационарном состоянии

$$x(0) = x_0,\ y(0) = y_0$$

то в любой момент времени численность популяций изменяться не будет. При малом отклонении от положения равновесия численности как хищника, так и жертвы с течением времени не возвращаются к равновесным значениям, а совершают периодические колебания вокруг стационарной точки. Амплитуда колебаний и их период определяется начальными значениями численностей $x(0)$, y(0). Колебания совершаются в противофазе.

5 Выполнение лабораторной работы

1. Напишем программу на Julia. Подкючим пакеты “Plots” и “DifferentialEquations”, объявим начальные данные. Далее объявим начальное условие для системы дифференциальных уравнений и промежуток времени, на котором будет проходить моделирование. После этого объявим функцию, представляющую систему. Построим график зависимости x от y и графики функций $x(t)$, $y(t)$. При помощи ‘DifferentialEquations’ зададим и решим систему ДУ, после чего построим графики функций $x(t)$, $y(t). Так же создадим два списка, в которых будут храниться точки уравнений. Воспользуемся данным списком, чтобы построить график зависимости x от y.

```
using Plots
using DifferentialEquations

a = -0.71
b = 0.046
c = 0.64
d = -0.017
T = (0.0, 50.0)
u0 = [4, 12]

function F!(du, u, p, t)
    du[1] = a*u[1] + b*u[1]*u[2]
    du[2] = c*u[2] + d*u[1]*u[2]
```

```
end

prob = ODEProblem(F!, u0, T)
sol = solve(prob, dtmax=0.05)

xx = []
yy = []
tt = sol.t

for u in sol.u
    x, y = u
    push!(xx, x)
    push!(yy, y)
end

plt = plot(layout=(1,2), dpi =150, size=(800,400), plot_title="Модель хищник-жерт

plot!(plt[1], tt, [xx, yy], color = [:red :blue], xlabel="time", ylabel = "x(t),
plot!(plt[2], yy, xx, color=:black, xlabel="y(t)", ylabel="x(t)")

savefig(plt, "lab5_1_jl.png")
```

2. В качестве результата получили график колебания изменения численности хищников и жертв, график зависимости изменения численности хищников от изменения численности жертв. (рис. 5.1)

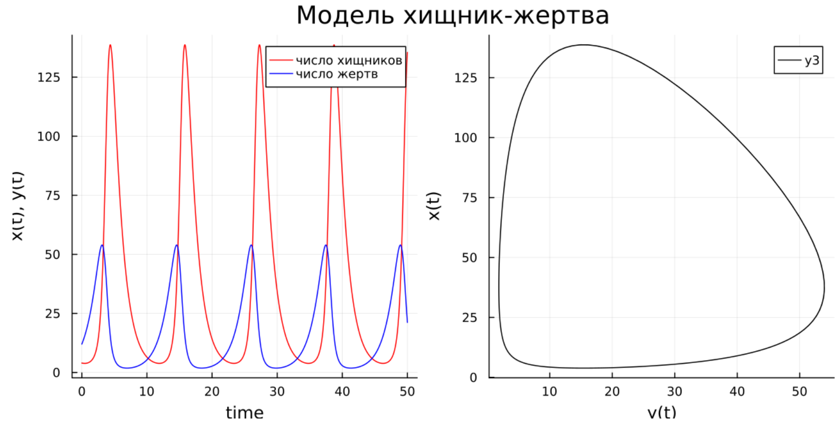

Рис. 5.1: Julia. Графики модели “Хищник-жертва” при $x_0 = 4, y_0 = 12$

3. Изменим начальные значения, при которых будет достигаться положение равновесия (не зависящее от времени решение). В качестве результата получим новые графики. (рис. 5.2)

```
u0 = [c/d, a/b]
```

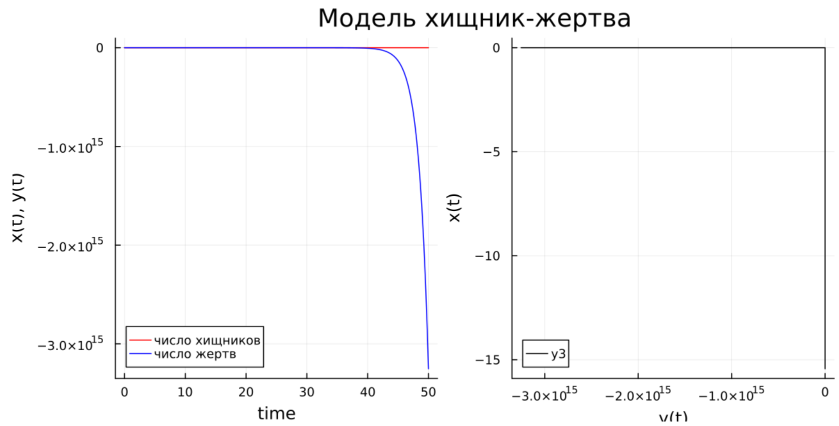

Рис. 5.2: Julia. Графики модели “Хищник-жертва” (стационарное состояние)

6. Построим график зависимости численности хищников от численности жертв, а также графики изменения численности хищников и численности жертв при начальных условиях $x_0 = 4, y_0 = 12$ на Modelica. (рис. 5.3, 5.4)

```
model lab5_1
parameter Real a = -0.71;
parameter Real b = 0.046;
parameter Real c = 0.64;
parameter Real d = -0.017;
Real x(start=4);
Real y(start=12);
equation
der(x) = a*x + b*x*y;
der(y) = c*y + d*x*y;
end lab5_1;
```

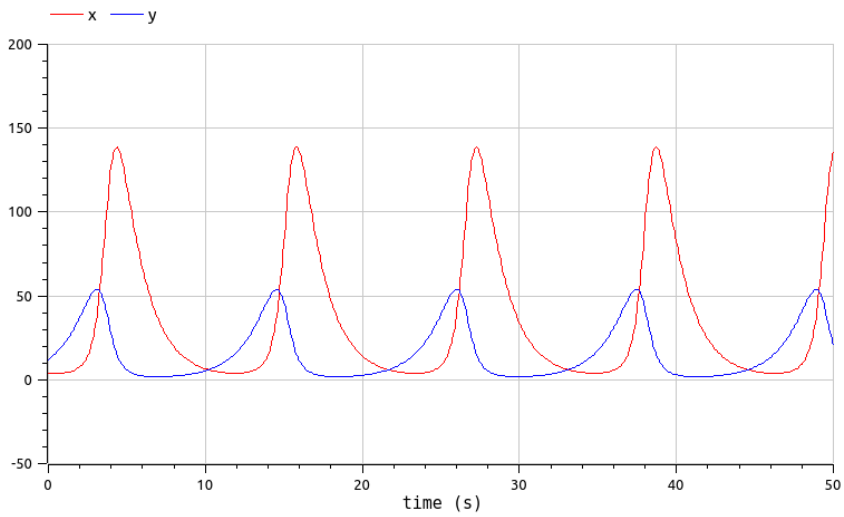

Рис. 5.3: Modelica. Графики функций изменения численности хищников и изменения численности жертв при $x_0 = 4, y_0 = 12$

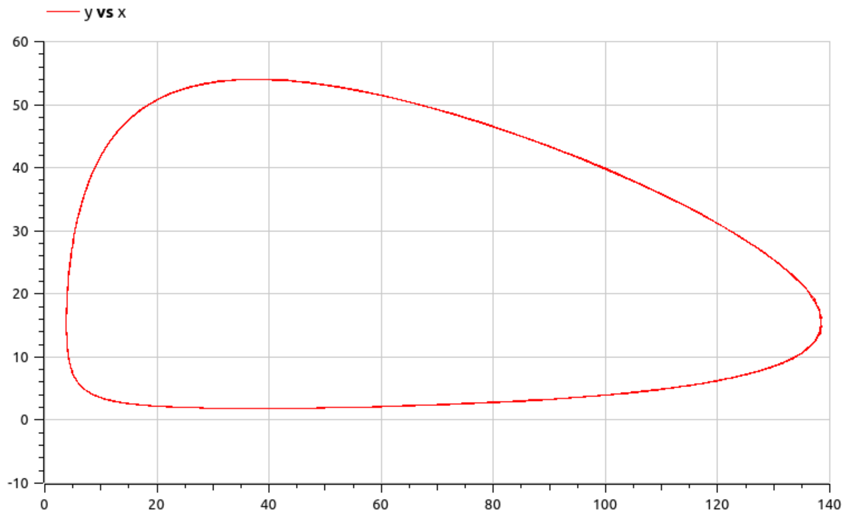

Рис. 5.4: Modelica. График зависимости изменения численности хищников от изменения численности жертв при $x_0 = 4, y_0 = 12$

7. Построим график зависимости численности хищников от численности жертв, а также графики изменения численности хищников и численности жертв в стационарном состоянии на Modelica. (рис. 5.5, 5.6)

```
model lab5_2
parameter Real a = -0.71;
parameter Real b = 0.046;
parameter Real c = 0.64;
parameter Real d = -0.017;
Real x(start=a/b);
Real y(start=c/d);
equation
der(x) = a*x + b*x*y;
der(y) = c*y + d*x*y;
end lab5_2;
```

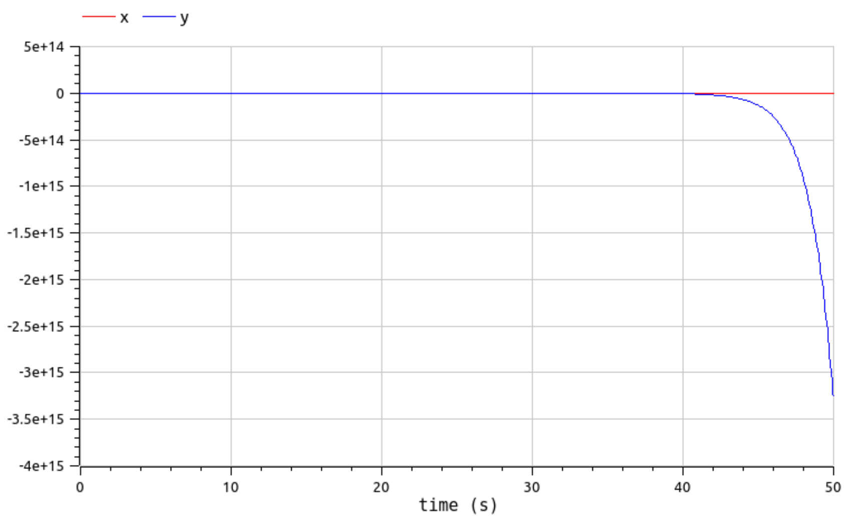

Рис. 5.5: Modelica. Графики функций изменения численности хищников и изменения численности жертв (стационарное состояние)

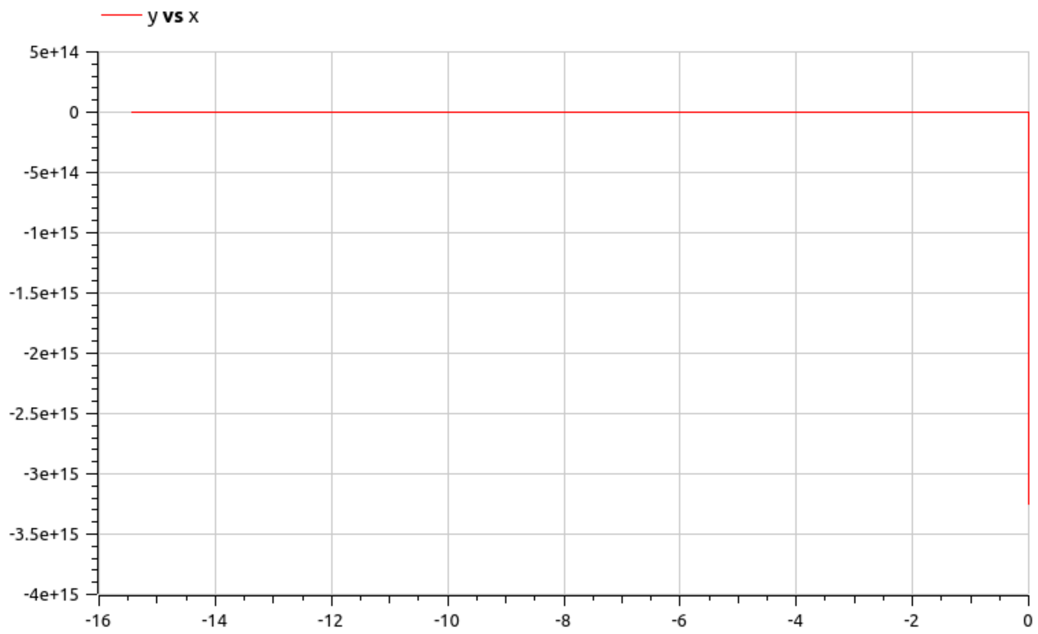

Рис. 5.6: Modelica. График зависимости изменения численности хищников от изменения численности жертв (стационарное состояние)

6 Анализ результатов

Моделирование на OMEdit оказалось в разы проще и быстрее, чем при использовании средств Julia. Скрипт на Modelica вышел более понятным и коротким. Более того OpenModelica быстрее обрабатывала скрипт и симмулировала модель. Стоит отметить, что OpenModelica имеет множество разлиных полезных инструментов для настройки с симмуляцией и работой с ней. К плюсам Julia можно отнести, что она является языком программирования, который хорошо подходит для математических и технических задач.

7 Выводы

Мы улучшили практические навыки в области дифференциальных уравнений, улучшили навыки моделирования на Julia, также навыки моделирования на OpenModelica. Изучили модель взаимодействия двух видов типа «хищник — жертва», а именно модель Лотки-Вольтерры.

Список литературы

1. Julia (язык программирования) [Электронный ресурс]. URL: https://w.wiki/6Ri4.
2. OpenModelica [Электронный ресурс]. URL: https://openmodelica.org/.
3. Модель хищник-жертва [Электронный ресурс]. URL: https://esystem.rudn.ru/mod/resource/view.php?id=967245.