Лабораторная работа №5

Модель хищник-жертва

Майсаров А.М.

Содержание

# 1 Цель работы

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

# 2 Задачи

Для модели «хищник-жертва»:

1. Построить график зависимости численности хищников от численности жертв, а также графики изменения численности хищников и численности жертв при следующих начальных условиях: , .
2. Найти стационарное состояние системы.

# 3 Среда

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

# 4 Теоретическое введение

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

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;

В этой модели – число жертв, - число хищников. Коэффициент 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 (). Каждый акт взаимодействия уменьшает популяцию жертв, но способствует увеличению популяции хищников (члены и в правой части уравнения).

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

Если начальные значения задать в стационарном состоянии

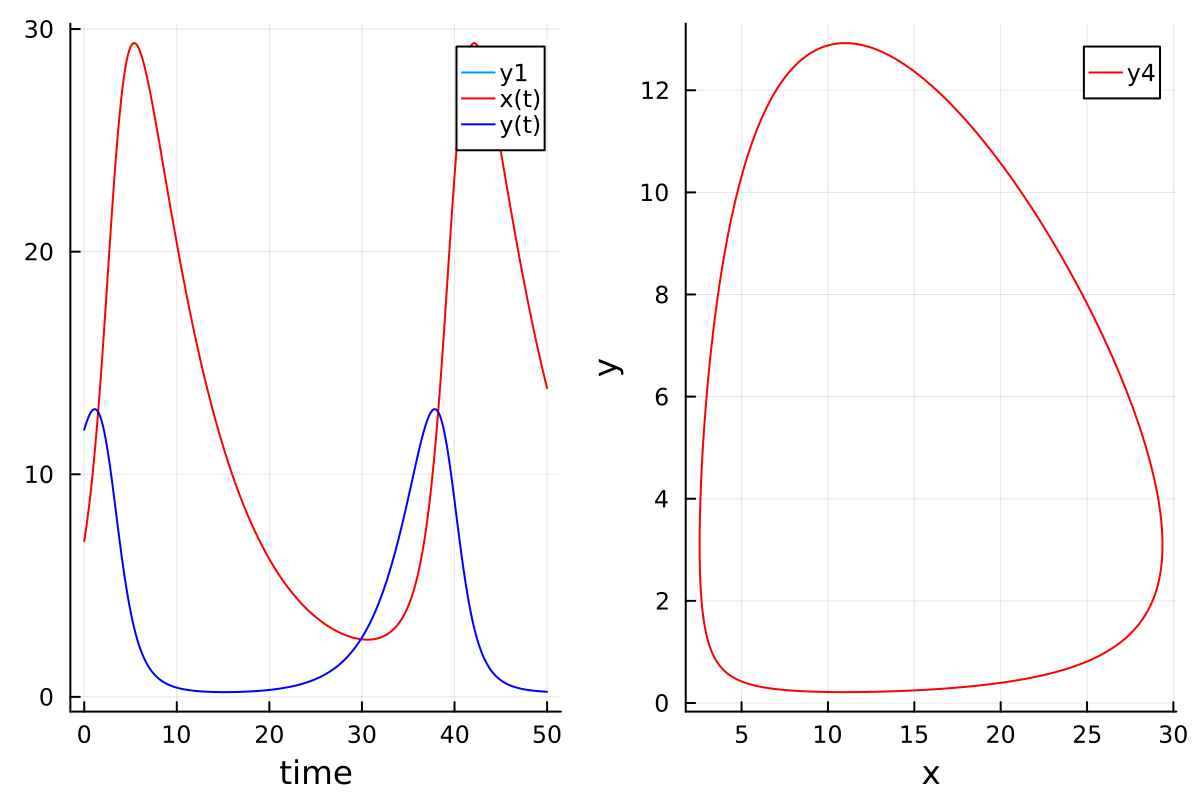
то в любой момент времени численность популяций изменяться не будет. При малом отклонении от положения равновесия численности как хищника, так и жертвы с течением времени не возвращаются к равновесным значениям, а совершают периодические колебания вокруг стационарной точки. Амплитуда колебаний и их период определяется начальными значениями численностей , 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 от и графики функций , . При помощи ‘DifferentialEquations’ зададим и решим систему ДУ, после чего построим графики функций , $y(t). Так же создадим два списка, в которых будут храниться точки уравнений. Воспользуемся данным списком, чтобы построить график зависимости от .

using Plots   
using DifferentialEquations  
  
a = 0.13  
b = 0.042  
c = 0.33  
d = 0.03  
  
u0 = [7, 12]  
T = (0, 50)  
  
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)  
  
X = []  
Y = []  
Time = sol.t   
  
for u in sol.u   
 x, y = u   
 push!(X, x)  
 push!(Y, y)  
end   
  
plt = plot(layout=(1,2), dpi = 200, size(800,400))  
plot!(plt[1], Time, [X, Y], color=[:red :blue], xlabel ="time", label = ["x(t)" "y(t)"])  
plot!(plt[2], X, Y, color=:red, xlabel="x", ylabel="y")  
  
savefig(plt, "lab5v1jl.png")

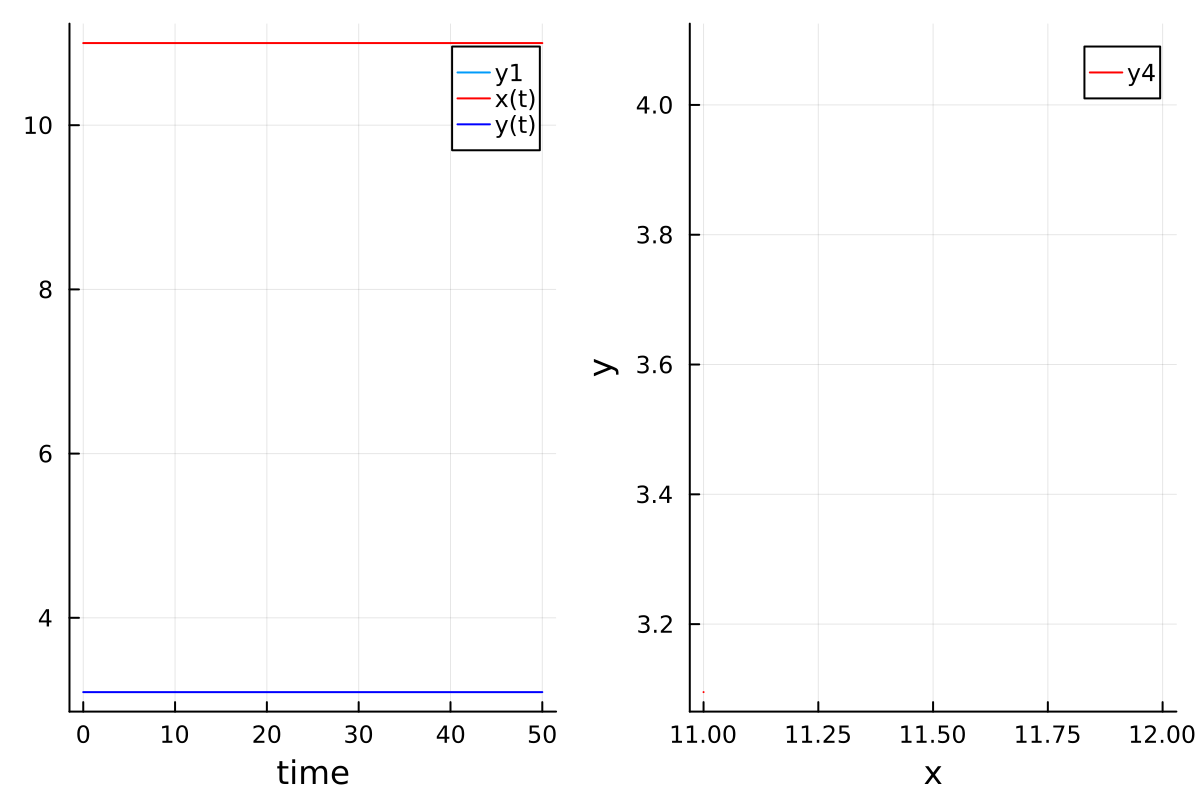
1. В качестве результата получили график колебания изменения численности хищников и жертв, график зависимости изменения численности хищников от изменения численности жертв. (рис. ??)



Julia. Графики модели “Хищник-жертва” при

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

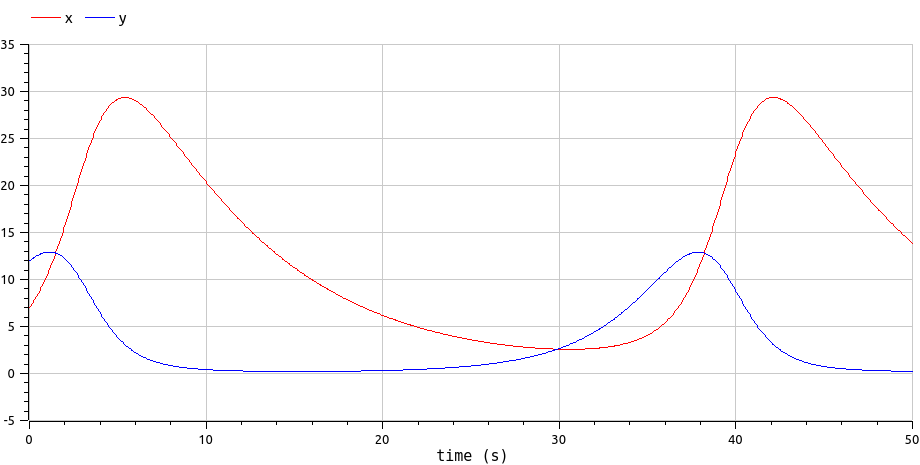
u0 = [c/d, a/b]



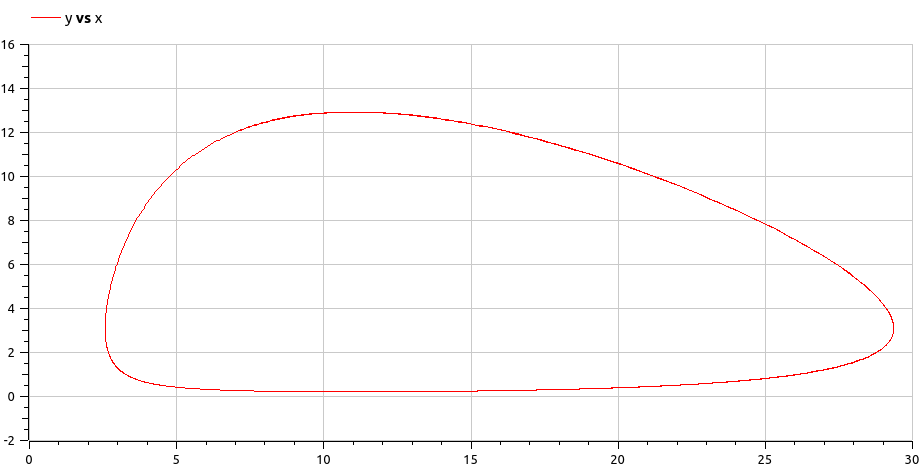
Julia. Графики модели “Хищник-жертва” (стационарное состояние)

1. Построим график зависимости численности хищников от численности жертв, а также графики изменения численности хищников и численности жертв при начальных условиях на Modelica. (рис. ??, ??)

model lab5v1  
parameter Real a = 0.13;  
parameter Real b = 0.042;  
parameter Real c = 0.33;  
parameter Real d = 0.03;  
Real x(start=7);  
Real y(start=12);  
equation  
der(x) = -a\*x + b\*x\*y;  
der(y) = c\*y -d\*x\*y;  
end lab5v1;



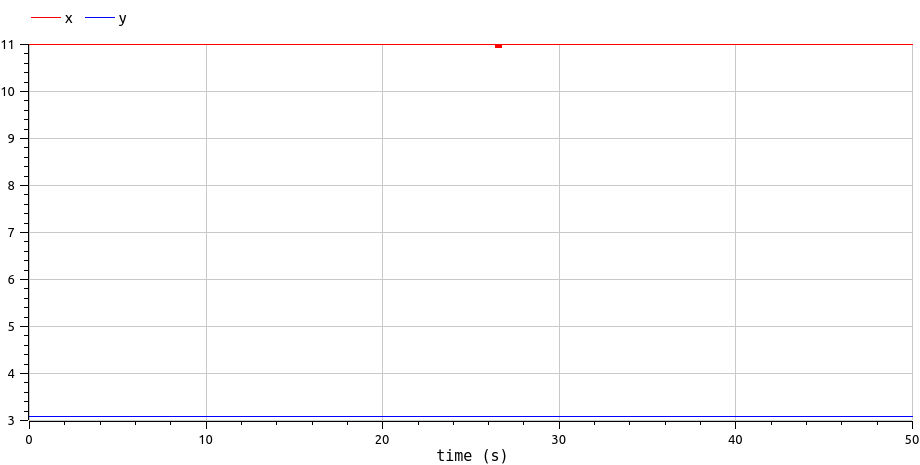
Modelica. Графики функций изменения численности хищников и изменения численности жертв при



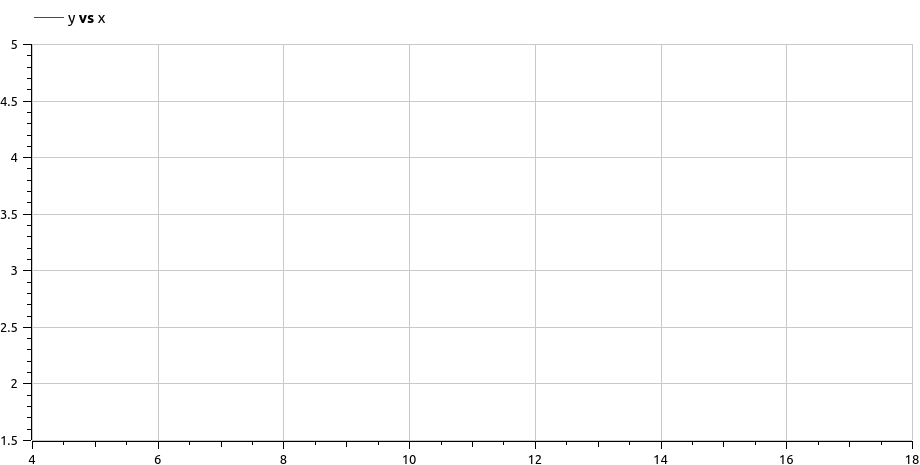
Modelica. График зависимости изменения численности хищников от изменения численности жертв при

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

model lab5v1  
parameter Real a = 0.13;  
parameter Real b = 0.042;  
parameter Real c = 0.33;  
parameter Real d = 0.03;  
Real x(start=c/d);  
Real y(start=a/b);  
equation  
der(x) = -a\*x + b\*x\*y;  
der(y) = c\*y -d\*x\*y;  
end lab5v1;



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



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