Лабораторная работа №3

Математическое моделирование

Лилия Руслановна Чекалова

Содержание

Список таблиц

Список иллюстраций

Цель работы

- Построение простейшей математической модели боевых действий — модели Ланчестера
- Визуализация модели на языках Julia и OpenModelica
- Сравнение языков Julia и OpenModelica

Задание

- Построить график изменения численности войск армии X и армии Y на основе модели боевых действий между регулярными войсками
- Построить график изменения численности войск армии X и армии Y на основе модели боевых действий с участием регулярных войск и партизанских отрядов
- Проанализировать графики

Теоретическое введение

Модель Ланчестера является простейшей моделью для описания боевых действий. Основной характеристикой соперников являются численности сторон (если какая-то из численностей обращается в ноль при положительной численности соперника, то данная сторона считается проигравшей).

Существует три случая ведения боевых действий:

- Боевые действия между регулярными войсками
- Боевые действия с участием регулярных войск и партизанских отрядов
- Боевые действия между партизанскими отрядами

В рамках нашей задачи мы будем рассматривать только первые два случая. Для описания этих случаев будут использоваться общие обозначения:

- a(t) и h(t) — коэффициенты, описывающие потери, не связанные с боевыми действиями (болезнь, дезертирство и пр.)
- b(t) и c(t) — коэффициенты, отражающие потери на поле боя
- P(t) и Q(t) — функции, учитывающие возможность подхода подкрепления к войскам

В первом случае (бой между регулярными войсками) модель имеет вид:

$$\begin{cases} \frac{\mathrm{d}x}{\mathrm{d}t} = -a(t)x(t) - b(t)y(t) + P(t) \\ \frac{\mathrm{d}y}{\mathrm{d}t} = -c(t)x(t) - h(t)y(t) + Q(t) \end{cases}$$

Во втором случае в борьбу добавляются партизанские отряды. Нерегулярные войска в отличии от постоянной армии менее уязвимы, так как действуют скрытно, в этом случае

сопернику приходится действовать неизбирательно, по площадям, занимаемым партизанами. Поэтому считается, что темп потерь партизан, проводящих свои операции в разных местах на некоторой известной территории, пропорционален не только численности армейских соединений, но и численности самих партизан. В результате модель принимает вид:

$$\begin{cases} \frac{\mathrm{d}x}{\mathrm{d}t} = -a(t)x(t) - b(t)y(t) + P(t) \\ \frac{\mathrm{d}y}{\mathrm{d}t} = -c(t)x(t)y(t) - h(t)y(t) + Q(t) \end{cases}$$

Более подробно о модели боевых действий см. в [1,2].

Выполнение лабораторной работы

Построив модель для случая боевых действий между регулярными войсками, пишем программу на языке Julia (рис. 0.1). Указываем начальные значения и коэффициенты, задаем функции возможности подхода подкрепления и функцию, описывающую нашу модель. С помощью библиотеки DifferentialEquations находим решение системы [3] и визуализируем его средствами библиотеки Plots.

```julia
using Plots
using DifferentialEquations

const x0 = 21000
const y0 = 9850

const a = 0.44
const b = 0.83
const c = 0.45
const h = 0.71

P(t) = cos(t) + 1
Q(t) = sin(t) + 1

u0 = [x0, y0]
p = (a, b, c, h)
T = (0, 1.5)

function F(du, u, p, t)
    a, b, c, h = p
    du[1] = -a*u[1] - b*u[2] + P(t)
    du[2] = -c*u[1] - h*u[2] + Q(t)
end

prob = ODEProblem(F, u0, T, p)

sol = solve(prob)

plt = plot(sol, vars=(0,1), color=:red, label="Армия x", title="Модель боевых действий №1", ylabel="Численность армии")
plot!(sol, vars=(0,2), color=:blue, label="Армия y", xlabel="Время")

savefig(plt, "lab3_1.png")
```

Рис. 0.1: Программа на Julia для первого случая

Из полученного графика можно сделать вывод, что армия Y в заданных условиях является проигравшей стороной, так как численность ее армии доходит до нуля (рис. 0.2).

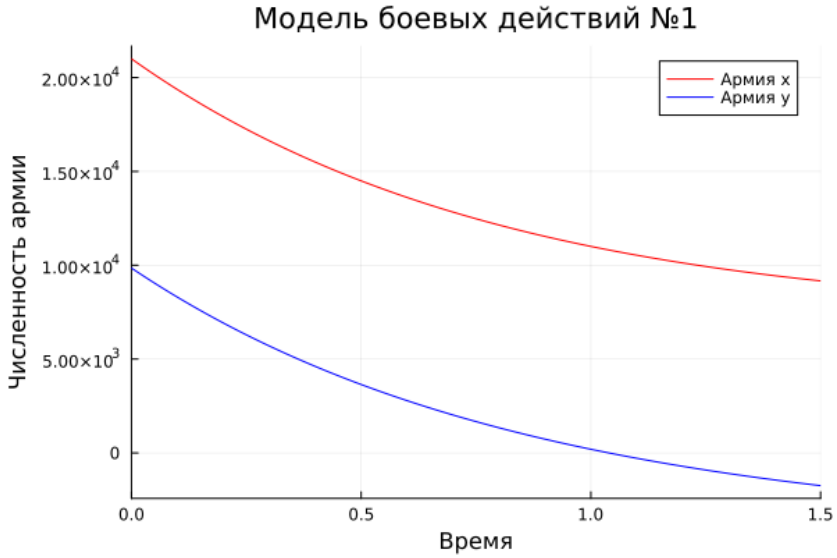

Рис. 0.2: График на Julia для первого случая

Меняем значения коэффициентов и функций P(t) и Q(t), а также слегка меняем функцию, описывающую нашу модель, чтобы она соответствовала второму случаю (рис. 0.3).

```julia
const a = 0.31
const b = 0.78
const c = 0.25
const h = 0.71

P(t) = abs(cos(2*t))
Q(t) = abs(sin(4*t))

u0 = [x0, y0]
p = (a, b, c, h)
T = (0, 0.001)

function F(du, u, p, t)
    a, b, c, h = p
    du[1] = -a*u[1] - b*u[2] + P(t)
    du[2] = -c*u[1]*u[2] - h*u[2] + Q(t)
end
```

Рис. 0.3: Программа на Julia для второго случая

Из графика видно, что армия Y стремительно теряет в численности при указанных условиях и опять проигрывает (рис. 0.4).

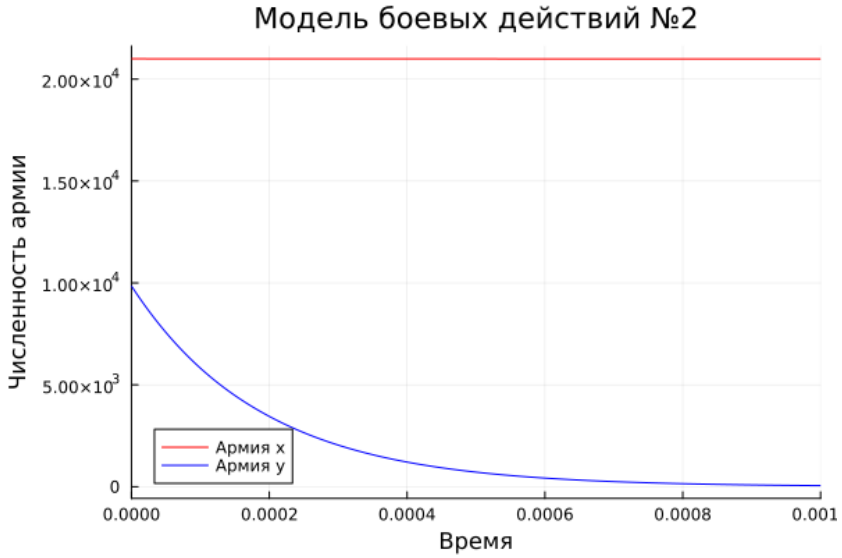

Рис. 0.4: График на Julia для второго случая

Далее описываем модель для первого случая на OpenModelica (рис. 0.5). В параметрах указываем начальные значения и коэффициенты, в разделе equation задаем функции, описывающие модель.

```
model Battle
parameter Integer x0 = 21000;
parameter Integer y0 = 9850;
parameter Real a = 0.44;
parameter Real b = 0.83;
parameter Real c = 0.45;
parameter Real h = 0.71;
Real P;
Real Q;
Real x(start=x0);
Real y(start=y0);
equation
P = cos(time) + 1;
Q = sin(time) + 1;
der(x) = -a * x - b * y + P;
der(y) = -c * x - h * y + Q;
end Battle;
```

Рис. 0.5: Программа на OpenModelica для первого случая

В установке симуляции настроим начальное и конечное время, а также размер интервала (рис. 0.6).

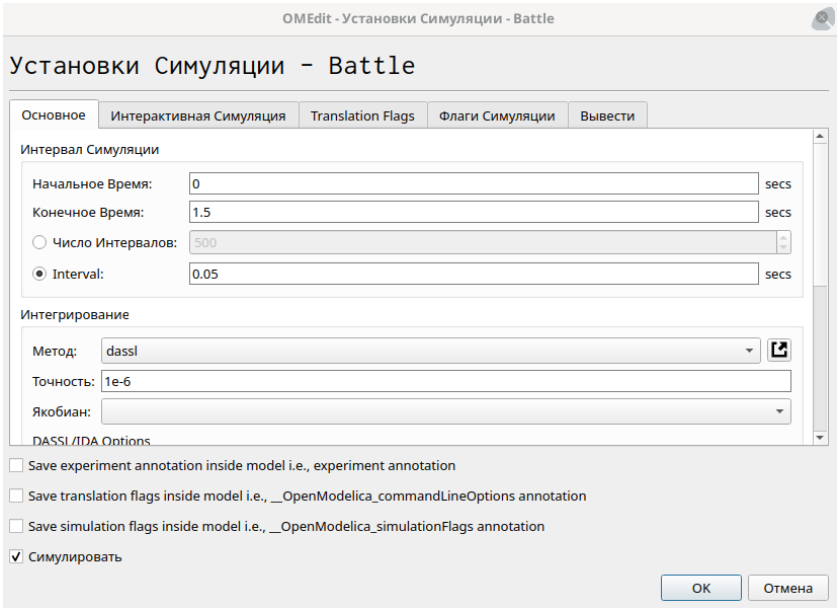

Рис. 0.6: Установка симуляции OpenModelica

Полученный график демонстрирует, что армия Y находится на проигрывающей позиции (рис. 0.7).

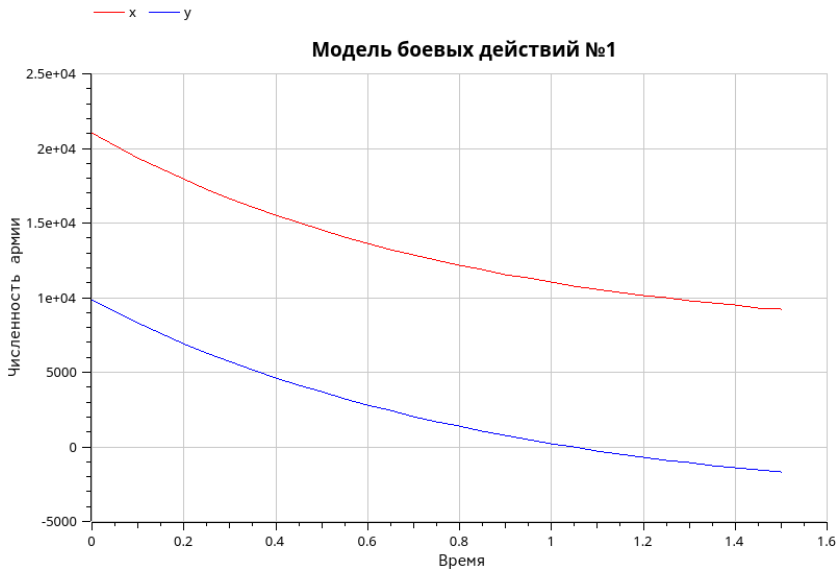

Рис. 0.7: График на OpenModelica для первого случая

Для второго случая меняем значения параметров-коэффициентов (рис. 0.8) и функции,

задающие модель (рис. 0.9).

```
parameter Real a = 0.31;
parameter Real b = 0.78;
parameter Real c = 0.25;
parameter Real h = 0.71;
```

Рис. 0.8: Изменение параметров на OpenModelica для второго случая

```
P = abs(cos(2*time));
Q = abs(sin(4*time));
der(x) = -a * x - b * y + P;
der(y) = -c * x * y - h * y + Q;
```

Рис. 0.9: Изменение функций на OpenModelica для второго случая

График показывает, что проигрывающей стороной является армия Y, чья численность значительно упала в короткие сроки (рис. 0.10).

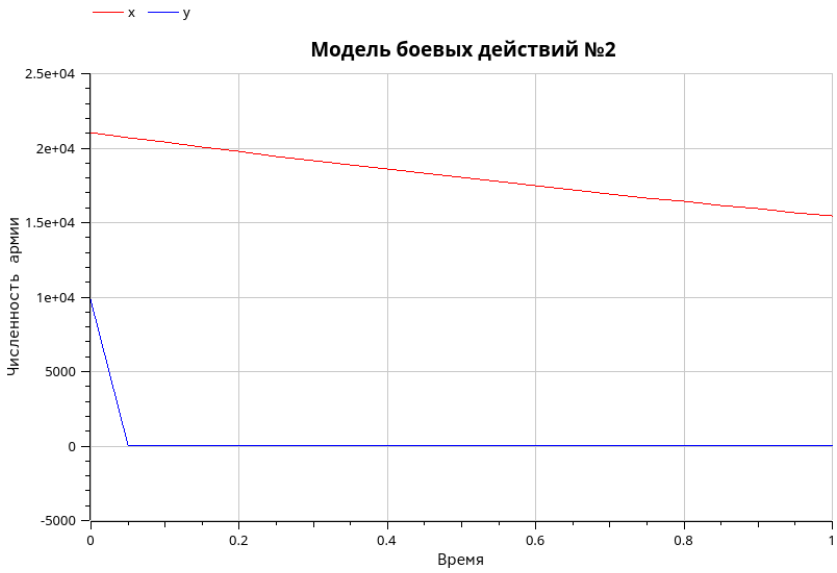

Рис. 0.10: График на OpenModelica для второго случая

Выводы

В ходе работы была освоена простейшая модель боевых действий — модель Ланчестера, и были применены навыки работы с Julia и OpenModelica для визуализации модели с помощью графиков. Результатом работы стали графики, демонстрирующие изменение численности двух армий на основе двух случаев ведения боевых действий, которые позволили нам сделать выводы о том, какая сторона считается проигравшей.

Сравнивая Julia и OpenModelica, отмечу, что, на мой взгляд, OpenModelica больше подходит для решения данной задачи, так как она специализируется на работе с дифференциальными уравнениями, в то время как Julia требует применения дополнительных библиотек.

Список литературы

1. Теоретические материалы к лабораторной работе ”Модель боевых действий” [Электронный ресурс]. URL: https://esystem.rudn.ru/mod/resource/view.php?id=967237.
2. Законы Осипова-Ланчестера [Электронный ресурс]. URL: https://ru.wikipedia.org/wiki/%D0%97%D0%B0%D0%BA%D0%BE%D0%BD%D1%8B_%D0%9E%D1%81%D0%B8%D0%BF%D0%BE%D0%B2%D0%B0_%E2%80%94_%D0%9B%D0%B0%D0%BD%D1%87%D0%B5%D1%81%D1%82%D0%B5%D1%80%D0%B0.
3. Решение ОДУ на Julia [Электронный ресурс]. URL: https://nextjournal.com/sosiris-de/ode-diffeq.