

Лабораторная работа №3

Математическое моделирование

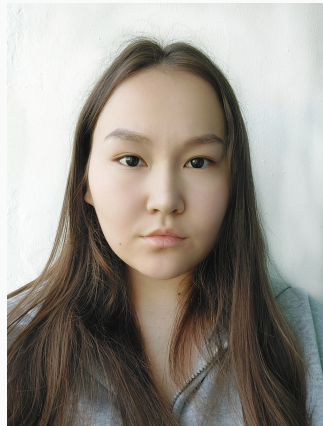
Гайсина А. Р.

20 апреля 2024

Российский университет дружбы народов, Москва, Россия

Информация

- Гайсина Алина Ринатовна
- Студентка группы НКНбд-01-21
- Студ. билет 1032216464
- Российский университет дружбы народов



Изучение модели боевых действий.

Вводная часть

Справка о языках программирования: Julia — высокоуровневый высокопроизводительный свободный язык программирования с динамической типизацией, созданный для математических вычислений. Эффективен также и для написания программ общего назначения. Синтаксис языка схож с синтаксисом других математических языков (например, MATLAB и Octave), однако имеет некоторые существенные отличия. Julia написан на Си, C++ и Scheme. Имеет встроенную поддержку многопоточности и распределённых вычислений, реализованные в том числе в стандартных конструкциях. [1]

Законы Ланчестера (законы Осипова — Ланчестера) — математическая формула для расчета относительных сил пары сражающихся сторон — подразделений вооруженных сил

Уравнения Ланчестера — это дифференциальные уравнения, описывающие зависимость между силами сражающихся сторон A и D как функцию от времени, причем функция зависит только от A и D .

В 1916 году, в разгар первой мировой войны, Фредерик Ланчестер разработал систему дифференциальных уравнений для демонстрации соотношения между противостоящими силами. Среди них есть так называемые Линейные законы Ланчестера (первого рода или честного боя, для рукопашного боя или неприцельного огня) и Квадратичные законы Ланчестера (для войн начиная с XX века с применением прицельного огня, дальнобойных орудий, огнестрельного оружия). В связи с установленным приоритетом в англоязычной литературе наметилась тенденция перехода от фразы «модель Ланчестера» к «модели Осипова — Ланчестера».

- В противоборстве могут принимать участие как регулярные войска, так и партизанские отряды. В общем случае главной характеристикой соперников являются численности сторон. Если в какой-то момент времени одна из численностей обращается в нуль, то данная сторона считается проигравшей (при условии, что численность другой стороны в данный момент положительна).

Рассматривается три случая ведения боевых действий: 1. Боевые действия между регулярными войсками 2. Боевые действия с участием регулярных войск и партизанских отрядов 3. Боевые действия между партизанскими отрядами

Задание (Вариант№35)

Между страной X и страной Y идет война. Численность состава войск исчисляется от начала войны, и являются временными функциями $x(t)$ и $y(t)$. В начальный момент времени страна X имеет армию численностью **31050** человек, а в распоряжении страны Y армия численностью в **20002** человек. Для упрощения модели считаем, что коэффициенты a, b, c, h - постоянны. Также считаем $P(t)$ и $Q(t)$ - непрерывные функции.

Постановка задачи

Постройте графики изменения численности войск армии X и армии Y для следующих случаев:

1. Модель боевых действий между регулярными войсками:

$$\begin{cases} \frac{dx}{dt} = -0.25x(t) - 0.74y(t) + \sin(t + 5) \\ \frac{dy}{dt} = -0.64x(t) - 0.55y(t) + \cos(t + 6) \end{cases}$$

. 2. Модель ведение боевых действий с участием регулярных войск и партизанских отрядов:

$$\begin{cases} \frac{dx}{dt} = -0.32x(t) - 0.89y(t) + 2\sin(10t) \\ \frac{dy}{dt} = -0.51x(t) - 0.62y(t) + 2\cos(10t) \end{cases}$$

.

Выполнение лабораторной работы

```
using Plots
using DifferentialEquations

people = Float64[31050, 20002]

# Первый случай
prom = [0.0, 2.0]
function du(du, u, p, t)
    du[1] = -0.25*u[1] - 0.74*u[2] + sin(t+5)
    du[2] = -0.64*u[1] - 0.55*u[2] + cos(t+6)
end
```

```
prob = ODEProblem(du, people, prom)
sol = solve(prob, dtmax=0.1)
A1 = [u[1] for u in sol.u]
A2 = [u[2] for u in sol.u]
T = [t for t in sol.t]
```

```
plt = plot(title="Модель боевых действий - случай 1", legend=:outerbottom)
plot!(plt, T, A1, label="Численность армии X")
plot!(plt, T, A2, label="Численность армии Y", color=:pink)
savefig("3_1.png")
```

```
# Второй случай
prom = [0.0, 1.0]
function du(du, u, p, t)
    du[1] = -0.32*u[1] - 0.89*u[2] + 2sin(10t)
    du[2] = -0.51*u[1] - 0.62*u[2] + 2cos(10t)
end
```

```
prob = ODEProblem(du, people, prom)
sol = solve(prob, dtmax=0.1)
A1 = [u[1] for u in sol.u]
A2 = [u[2] for u in sol.u]
T = [t for t in sol.t]
```

```
plt = plot(title="Модель боевых действий - случай 2", legend=:outerbottom)
```

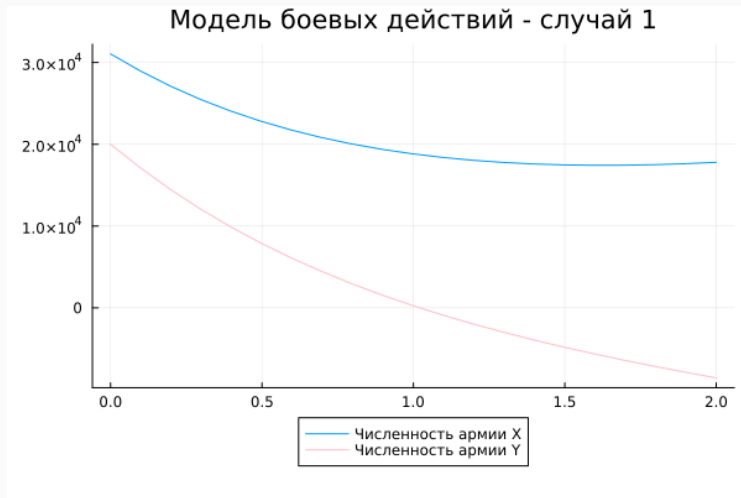


Рис. 1: Модель боевых действий между регулярными войсками

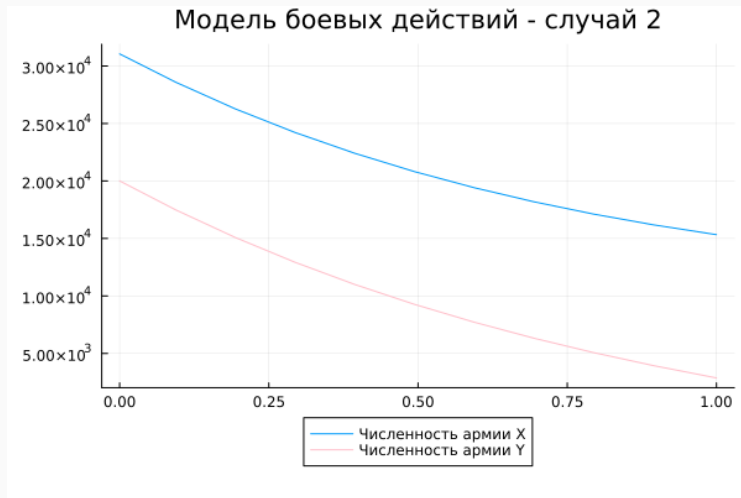


Рис. 2: Модель ведения боевых действий с участием регулярных войск и партизанских отрядов

```
Real y11(start=31050);  
Real y21(start=20002);  
parameter Real a = 0.25;  
parameter Real b = 0.74;  
parameter Real c = 0.64;  
parameter Real d = 0.55;
```

```
Real y12(start=31050);  
Real y22(start=20002);  
parameter Real a2 = 0.32;  
parameter Real b2 = 0.89;  
parameter Real c2 = 0.51;  
parameter Real d2 = 0.55;
```

equation

$$\text{der}(y_{11}) = -a*y_{11} - b*y_{21} + \sin(\text{time}+5);$$
$$\text{der}(y_{21}) = -c*y_{11} - d*y_{21} + \cos(\text{time}+6);$$

equation

$$\text{der}(y_{12}) = -a_2*y_{12} - b_2*y_{22} + 2*\sin(10*\text{time});$$
$$\text{der}(y_{22}) = -c_2*y_{12}*y_{22} - d_2*y_{22} + 2*\cos(10*\text{time});$$

При помощи языков программирования Julia и OpenModelica я построила 2 модели боевых действий.

При помощи языков программирования Julia и OpenModelica я построила 2 модели боевых действий.

[1] Документация по Julia: <https://docs.julialang.org/en/v1/> [2] Решение дифференциальных уравнений: <https://www.wolframalpha.com/>