

Лабораторная работа №2

Математическое моделирование

Чекалова Л. Р.

17 февраля 2023

Российский университет дружбы народов, Москва, Россия

- Чекалова Лилия Руслановна
- студент 3 курса группы НФИбд-02-20
- ст. б. 1032201654
- Российский университет дружбы народов
- 1032201654@pfur.ru

Вводная часть

- Необходимость решения задач поиска
- Необходимость визуализировать данные
- Простота построения моделей

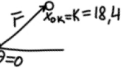

- Решить задачу о погоне с помощью дифференциальных уравнений
- Построить математическую модель задачи
- Визуализировать модель с помощью Julia и OpenModelica
- Проанализировать полученные результаты

- Средства языка Julia для визуализации данных
- GUI OMEdit для визуализации данных на OpenModelica
- Результирующие форматы
 - jl
 - mo
 - png

Ход работы

Подготовка к выполнению работы

Проведя ряд рассуждений, получаем дифференциальное уравнение.


$$\begin{aligned} \frac{x}{v} &= \frac{K-x}{4.6v} - I & x_1 &= \frac{K}{5.6} = \frac{18.4}{5.6} \\ \frac{x}{v} &= \frac{K+x}{4.6v} - II & x_2 &= \frac{K}{3.6} = \frac{18.4}{3.6} \end{aligned} \Rightarrow$$

$$\begin{aligned} v_r &= \frac{dr}{dt} = v \\ v_\theta &= \frac{d\theta}{dt} r \end{aligned}$$
$$v_\theta = \sqrt{21.16v^2 - v^2} = \sqrt{20.16} v = r \frac{d\theta}{dt}$$
$$\begin{cases} \frac{dr}{dt} = v \\ r \frac{d\theta}{dt} = \sqrt{20.16} v \end{cases} \quad \begin{matrix} \text{н.у.1} & \text{н.у.2} \\ \left\{ \begin{matrix} r_0 = x_1 \\ \theta_0 = 0 \end{matrix} \right\} & \left\{ \begin{matrix} r_0 = x_2 \\ \theta_0 = -\pi \end{matrix} \right\} \end{matrix}$$
$$\Downarrow$$
$$r \frac{d\theta}{dt} = \sqrt{20.16} \frac{dr}{dt} \Rightarrow \frac{d\theta}{r} = \frac{d\theta}{\sqrt{20.16}}$$

Решение дифференциального уравнения

$$\int \frac{dr}{r} = \int \frac{d\theta}{\sqrt{20.16}} \Rightarrow \ln r = \frac{\theta}{\sqrt{20.16}} + C$$

$$e^{\ln r} = e^{\frac{\theta}{\sqrt{20.16}} + C} = e^{\frac{\theta}{\sqrt{20.16}}} \cdot e^C$$

$$r = C e^{\frac{\theta}{\sqrt{20.16}}} \Rightarrow r(\theta) = C e^{\frac{\theta}{\sqrt{20.16}}}$$

$$r(0) = r_0$$

$$r_0 = C e^{\frac{0}{\sqrt{20.16}}} \Rightarrow C = r_0 = X_1$$

$$r(\theta) = X_1 e^{\frac{\theta}{\sqrt{20.16}}} - I$$

$$r(-\pi) = r_0$$

$$r_0 = C e^{\frac{-\pi}{\sqrt{20.16}}} \Rightarrow C = \frac{X_2}{e^{\frac{-\pi}{\sqrt{20.16}}}}$$

$$r(\theta) = \frac{X_2}{e^{\frac{-\pi}{\sqrt{20.16}}}} e^{\frac{\theta}{\sqrt{20.16}}} - II$$

Программа на языке Julia, 1

```
using Plots

fi = 3pi/4
fi0_1 = 0
r0_1 = 18.4/5.6
fi0_2 = -pi
r0_2 = 18.4/3.6

function F1(theta, r0, fi0)
    return r0/exp(fi0/sqrt(20.16))*exp(theta/sqrt(20.16))
end

th = collect(0:0.01:2pi)
rth = F1.(th, r0_1, fi0_1)

r = F1.(fi, r0_1, fi0_1)

x = r*cos(fi)
y = r*sin(fi)

@show x
@show y

plt = plot(
    proj=:polar,
    aspect_ratio=:equal,
    dpi=300,
    legend=true,
    title="Задача о погоне"
)

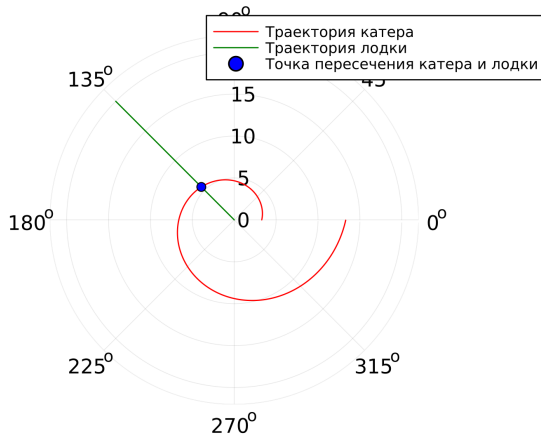
plot!(plt,
    th,
    rth,
    xlabel="theta",
    ylabel="r(theta)",
    color=:red,
    label="Траектория катера")

plot!(plt,
    [fi, fi],
    [0, 20],
    xlabel="fi",
    ylabel="r",
    color=:green,
    label="Траектория лодки")
```

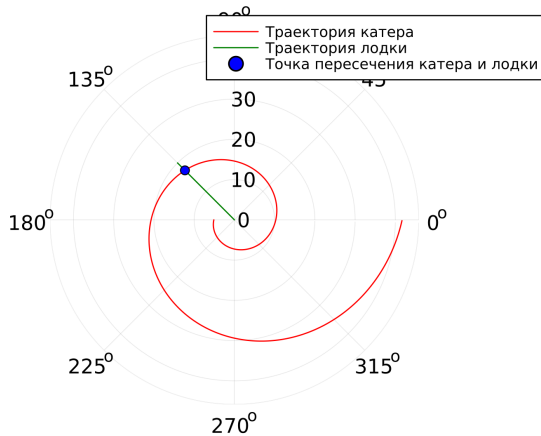
```
plot!(plt,  
[fi],  
[r],  
seriestype=:scatter,  
color=:blue,  
label="Точка пересечения катера и лодки")  
  
savefig(plt, "lab2_1.png")
```

```
th = collect(-pi:0.01:2pi)  
rth = F1.(th, r0_2, fi0_2)  
r = F1.(fi, r0_2, fi0_2)
```

Задача о погоне



Задача о погоне



Программа на языке OpenModelica

```
model Pursuit
parameter Real a=sqrt(20.16);
parameter Real k1= 18.4/5.6;
parameter Real k2 = 18.4/3.6;
constant Real pi = 3.14;
Real theta(start=-pi/2);
Real r(start=k1);
// Real r(start=k2);
// Real theta(start=-3*pi/2);
Real x(start=0);
Real y(start=0);
Real r1(start=0);
Real f1(start=0);
equation
x = time;
y = -x;
der(r) = 1;
der(theta) = a / r;
r1 = r*cos(theta);
f1 = -r*sin(theta);
end Pursuit;
```

График на языке OpenModelica, 1 случай

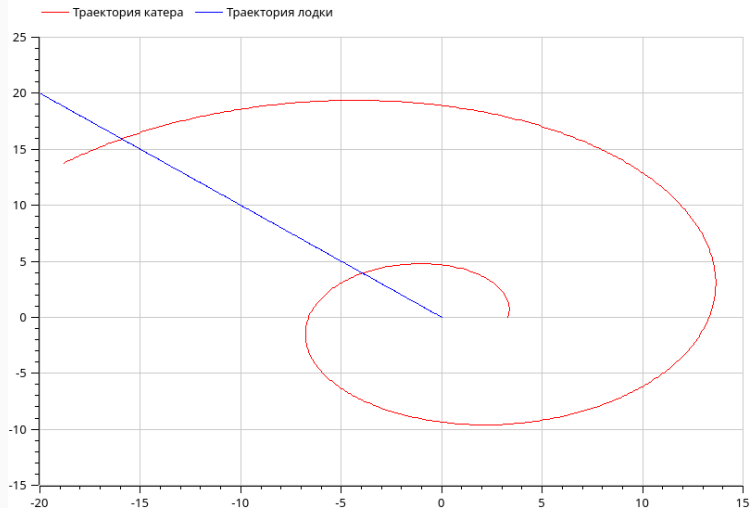
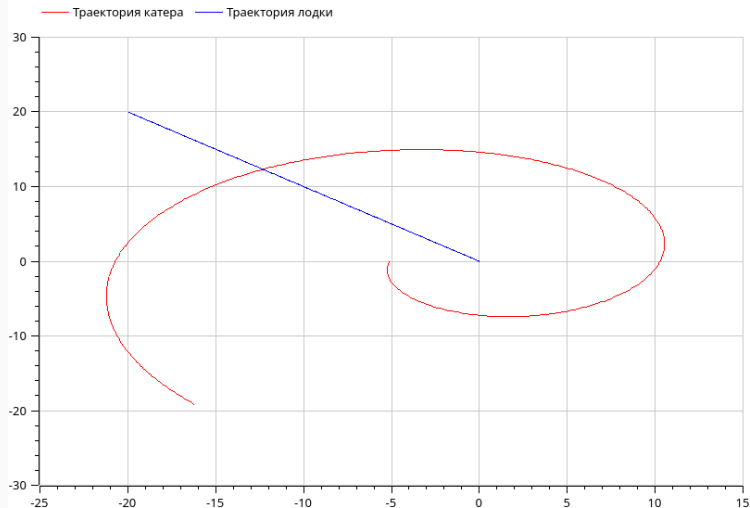


График на языке OpenModelica, 2 случай



Результаты

- Julia
 - более интуитивно понятный язык
 - позволяет производить различные вычисления, не ограничиваясь работой с визуализацией уравнений
- OpenModelica
 - более узконаправленный инструмент
 - ориентируется на создание сложных моделей с множеством уравнений в основе
 - работа с ним кажется труднее

- Приобретены навыки работы с Julia и OpenModelica
- Построена математическая модель задачи о погоне
- Построены графики, визуализирующие решение задачи

**Если долго мучиться, что-нибудь
получится. © А. Пугачева**
