

# Презентация к лабораторной работе 2

## Задача о погоне

---

Саттарова В. В.

18 февраля 2023

Российский университет дружбы народов, Москва, Россия

# **Вводная часть**

---

Задача о погоне является хорошим и наглядным примером построения математических моделей для выбора правильной стратегии при решении задач поиска с использованием полярных координат и языков научного программирования.

- Задача о погоне

- Вывести дифференциальное уравнение для решения задачи о погоне
- Написать код на Julia для вычисления уравнения и визуализации результатов

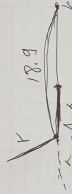
- Лист бумаги с ручкой
- Julia (REPL)
- Jupiter Notebook (IJulia)
- Курс на ТУИС “Математическое моделирование”

## **Содержание исследования**

---

# Вывод дифференциального уравнения

$$v_k = 5,5$$



$$t = \frac{Rt}{v_k} = \frac{18,9 - r}{5,5 v_k} \quad (1) \Rightarrow t = \frac{18,9 - r}{5,5}$$

$$\Rightarrow \frac{18,9 + r}{v_k} = \frac{18,9 + r}{5,5 v_k} \quad (2)$$

$$\Downarrow$$

$$18,9 - r = 5,5 t$$

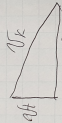
$$\Downarrow$$

$$18,9 + r = 5,5 t$$

$$\Downarrow$$

$$4,5 t = 18,9$$

$$t_2 = 4,2$$



$$v_k^2 = v_r^2 + v_t^2$$

$$(5,5 v_k)^2 = v_r^2 + v_t^2$$

$$30,25 v_k^2 = v_r^2 + v_t^2$$

$$v_t = \sqrt{30,25 v_k^2 - v_r^2}$$

$$\frac{dr}{dt} = v_r = \frac{r}{R} \frac{d\theta}{dt} = \frac{r}{R} \frac{d\theta}{dt}$$

$$\frac{dt}{dt} = \frac{dr}{v_r} \quad \frac{dt}{dt} = \frac{r d\theta}{\sqrt{30,25 v_k^2 - v_r^2}}$$

$$\Downarrow$$

$$\frac{dr}{d\theta} = \frac{r}{\sqrt{30,25}}$$



# Решение уравнения

← → ↻ http://localhost:8888/notebooks/Untitled2.ipynb?kernel\_name=julia-1.8



📄 + 🔍 📄 📄 ⬆ ⬆ ▶ Запуск 🛑 ↺ ⬆ Код

```
Ввод [8]: # Решаем ОДУ

using DifferentialEquations
using Plots

"""Правая часть ОДУ
u --- переменная (скаляр или массив)
p --- параметры (кортеж, tuple)
t --- аргумент (скаляр, время)
"""
function F(u, p, t)
    # p и t не используются в нашем примере
    # но нужны в любом случае для библиотеки
    return u / √(29.25)
end

"Начальное значение"
const u_0 = 2.9
const u_1 = 4.2

"Интервал (кортеж, tuple)"
const T0 = (0, 2π)
const T1 = (π, 3π)

# Задача
prob0 = ODEProblem(F, u_0, T0)
prob1 = ODEProblem(F, u_1, T1)

# Решение задачи
sol0 = solve(
    prob0,
    dtmax=0.1
)
sol1 = solve(
    prob1,
    dtmax=0.1
)
```

# Данные лодки и точка пересечения

```
Ввод [13]: boatr = Float64[0.0, 10.0]
            boatang = Float64[3π/2]

            point0 = 0
            point1 = 0
            for (i,ang) in enumerate(sol0.t)
                if (round(ang, digits=1) == round(3π/2, digits=1))
                    global point0 = sol0.u[i]
                end
            end

            for (i,ang) in enumerate(sol1.t)
                if (round(ang, digits=1) == round(3π/2, digits=1))
                    global point1 = sol1.u[i]
                end
            end
```

Рис. 3: Код

# Визуализация результатов

```
Ввод [11]: plt0 = plot(
    proj = :polar,
    aspect_ratio=:equal,
    dpi=300,
    title="Погоня 1",
    legend=true)

# воскл. знак в названии обязателен
# в данном случае!
plot!(
    plt0,
    sol0.t,
    sol0.u,
    xlabel="θ",
    ylabel="r(t)",
    label="Траектория катера",
    color=:red)
plot!(
    plt0,
    boatang,
    boatr,
    xlabel="θ",
    ylabel="r(t)",
    label="Траектория лодки",
    color=:blue)
plot!(
    plt0,
    boatang,
    [point0],
    seriestype = :scatter,
    label="Пересечение",
    color=:black)
```

# График 1

Out[11]:

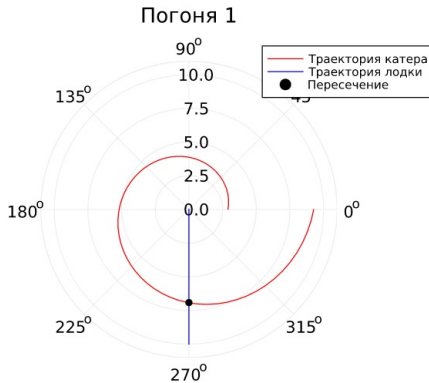


Рис. 5: 1

## График 2

Out[15]:



## Результаты

---

## Результаты работы

---

- Решена задача о погоне
- Построены графики