

# **Лабораторная работа №2**

**Задача о погоне**

Карымшаков Артур Алишерович

# Содержание

1	Цель работы	5
2	Задание	6
3	Выполнение лабораторной работы	7
4	Выводы	13

## Список таблиц

## Список иллюстраций

3.1	Начальные условия и уравнения, представленные в коде программы (1 часть) . . . . .	8
3.2	Начальные условия и уравнения, представленные в коде программы (2 часть) . . . . .	9
3.3	Траектории для 1го случая . . . . .	10
3.4	Траектории для 2го случая . . . . .	11
3.5	Координаты точки пересечения для 1го случая . . . . .	11
3.6	Координаты точки пересечения для 2го случая . . . . .	12

# 1 Цель работы

Построение математической модели для выбора правильной стратегии при решении задачи о погоне.

## 2 Задание

1. Записать уравнение, описывающее движение катера, с начальными условиями для двух случаев (в зависимости от расположения катера относительно лодки в начальный момент времени).
2. Построить траекторию движения катера и лодки для двух случаев.
3. Найти точку пересечения траектории катера и лодки

### 3 Выполнение лабораторной работы

1. Известны начальные данные задачи: лодка обнаруживается на расстоянии  $s = 25$  км от катера, и скорость катера в 5.1 раза больше чем скорость лодки. Для того, чтобы описать уравнение движения катера необходимо, во-первых, выразить

$$x_1 = \frac{s}{n + 1},$$

где ( $x_1$  - начальное расстояние между лодкой и катером для 1го случая)

$$x_2 = \frac{s}{n - 1},$$

где ( $x_2$  - начальное расстояние между лодкой и катером для 2го случая)

Во-вторых, нужно выразить дифференциальное уравнение в общем виде

$$x_1 = \frac{dr}{d\theta} = \frac{r}{\sqrt{n^2 - 1}}$$

Ниже представлен код программы, в котором присутствуют два вышеописанных уравнения. Часть 1: (рис 1. @fig:001)

```

import math
import numpy as np
from scipy.integrate import odeint
import matplotlib.pyplot as plot

n = 5.1
s = 25 # начальное расстояние от лодки до катера
fi = 3 * math.pi / 4

# функция, описывающая движения катера береговой охраны
def f(r, tetha):
    dr = r/math.sqrt(n*n - 1)
    return dr

# начальные условия:

# начальные для 1го случая:
r0 = s/(n+1)

# начальные для 2го случая:
#r0 = s/(n-1)

# для 1го случая:
tetha = np.arange(0,2*math.pi, 0.01)

# для 2го случая:
#tetha = np.arange(-math.pi,math.pi, 0.01)

r = odeint(f, r0, tetha)

```

Рис. 3.1: Начальные условия и уравнения, представленные в коде программы (1 часть)

Ниже представлен код программы, в котором присутствуют два вышеописанных уравнения. Часть 2: (рис 2. @fig:001)



```

r = odeint(f, r0, tetha)

# функция, описывающая движение лодки браконьеров
def f2(t):
    xt = math.tan(fi) * t
    return xt

t = np.arange(0.000001,20,1)

r1 = np.sqrt(t * t + f2(t) * f2(t))

tetha1 = np.arctan(f2(t)/t)

#Построение графиков функций
plot.polar(tetha, r, 'b') #движение катера охранников
plot.polar(tetha1, r1, 'r') #движение лодки браконьеров

tmp = 0
for i in range(len(tetha)):
    #if round(tetha[i],2) == round(fi + math.pi, 2): #для 1го случая
    if round(tetha[i],2) == round(fi - math.pi, 2): #для 2го случая
        tmp = i
#Координаты точки пересечения графиков движения охранников и браконьеров
#в полярных координатах
print("tetha = ", tetha[tmp], "and r = ", r[tmp][0])
#в декартовых координатах
print("x = ", r[tmp][0]/math.sqrt(2), "y = ", -r[tmp][0]/math.sqrt(2))

tetha = 0.0 and r = 4.0983606557377055
x = 2.897978611420277 y = -2.897978611420277

```

Рис. 3.2: Начальные условия и уравнения, представленные в коде программы (2 часть)

2. Добавлю в код программы функцию, позволяющую строить в полярных координатах траектории движений катера и лодки для 1го случая. Результат следующий: (рис 3. @fig:001)

---

```
tetha = 0.0 and r = 4.0983606557377055  
x = 2.897978611420277 y = -2.897978611420277
```

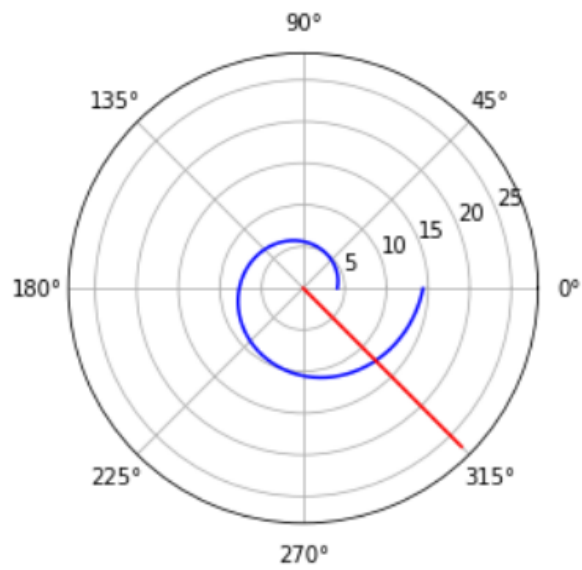


Рис. 3.3: Траектории для 1го случая

3. Теперь произведу изменения так, чтобы функция смогла построить в полярных координатах траектории движений катера и лодки для 2го случая (рис 4. @fig:001)

```
tetha = -0.7915926535898432 and r = 9.755145412443204
x = 6.897929472599429 y = -6.897929472599429
```

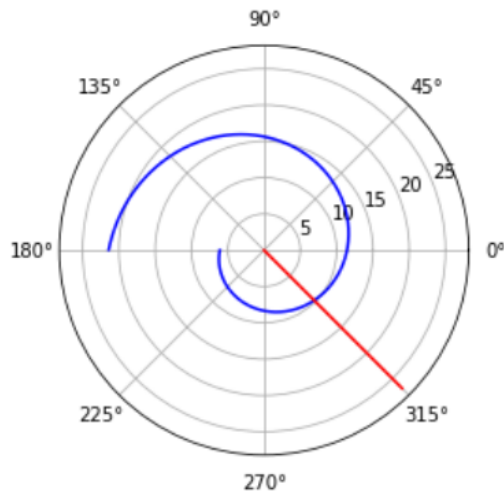


Рис. 3.4: Траектории для 2го случая

4. Найду точку пересечения траекторий катера и лодки для 1го случая. Точка пересечения данных графиков - точка, в которой радиусы и углы обоих функций совпадают (в полярных координатах).  
Добавлю фрагмент к коду программе, позволяющий найти эту точку и вывести координаты этой точки в полярных и декартовых координатах. Результат следующий: (рис 5. @fig:001)

---

```
tetha = 0.0 and r = 4.0983606557377055
x = 2.897978611420277 y = -2.897978611420277
```

Рис. 3.5: Координаты точки пересечения для 1го случая

5. Теперь найду координаты 2ой точки пересечения в декартовых и полярных координатах. Результат следующий: (рис 6. @fig:001)

```
tetha = -0.7915926535898432 and r = 9.755145412443204  
x = 6.897929472599429 y = -6.897929472599429
```

Рис. 3.6: Координаты точки пересечения для 2го случая

## 4 Выводы

Научился строить модель для выбора правильной стратегии при решении задачи о погоне.