Отчет по лабораторной работе №2

Задача о погоне - вариант 30

Тимур Еюбоглу

Содержание

# 1 Цель работы

Приведем один из примеров построения математических моделей для выбора правильной стратегии при решении задач поиска. Например, рассмотрим задачу преследования браконьеров береговой охраной. На море в тумане катер береговой охраны преследует лодку браконьеров. Через определенный промежуток времени туман рассеивается, и лодка обнаруживается на расстоянии k км от катера. Затем лодка снова скрывается в тумане и уходит прямолинейно в неизвестном направлении. Известно, что скорость катера в n раза больше скорости браконьерской лодки. Необходимо определить по какой траектории необходимо двигаться катеру, чтоб нагнать лодку.

# 2 Задание

1. Провести необходимые рассуждения и вывод дифференциальных уравнений, если скорость катера больше скорости лодки в n раз.
2. Построить траекторию движения катера и лодки для двух случаев.
3. Определить по графику точку пересечения катера и лодки.

# 3 Выполнение лабораторной работы

Принимаем за - место нахождения лодки браконьеров в момент обнаружения, - место нахождения катера береговой охраны относительно лодки браконьеров в момент обнаружения лодки.

Введем полярные координаты. Считаем, что полюс - это точка обнаружения лодки браконьеров , а полярная ось r проходит через точку нахождения катера береговой охраны.

Чтобы найти расстояние (расстояние после которого катер начнет двигаться вокруг полюса), необходимо составить простое уравнение. Пусть через время катер и лодка окажутся на одном расстоянии от полюса. За это время лодка пройдет , а катер (или , в зависимости от начального положения катера относительно полюса). Время, за которое они пройдут это расстояние, вычисляется как или (для второго случая ). Так как время одно и то же, то эти величины одинаковы. Тогда неизвестное расстояние можно найти из следующего уравнения: - в первом случае, во втором случае.

Отсюда мы найдем два значения и , задачу будем решать для двух случаев.

,при

,при

После того, как катер береговой охраны окажется на одном расстоянии от полюса, что и лодка, он должен сменить прямолинейную траекторию и начать двигаться вокруг полюса удаляясь от него со скоростью лодки . Для этого скорость катера раскладываем на две составляющие: - радиальная скорость и - тангенциальная скорость. Радиальная скорость - это скорость, с которой катер удаляется от полюса . Нам нужно, чтобы эта скорость была равна скорости лодки, поэтому полагаем . Тангенциальная скорость – это линейная скорость вращения катера относительно полюса. Она равна произведению угловой скорости на радиус , Найдем тангенциальную скорость для нашей задачи . Вектора образуют прямоугольный треугольник, откуда по теореме Пифагора можно найти тангенциальную скорость . Поскольку, радиальная скорость равна , то тангенциальную скорость находим из уравнения . Следовательно, .

Тогда получаем

Решение исходной задачи сводится к решению системы из двух дифференциальных уравнений

с начальными условиями

Исключая из полученной системы производную по t, можно перейти к следующему уравнению:

Начальные условия остаются прежними. Решив это уравнение, мы получим траекторию движения катера в полярных координатах. Теперь, когда нам известно все, что нам нужно, построим траекторию движения катера и лодки для двух случаев.

## 3.1 Условие задачи

На море в тумане катер береговой охраны преследует лодку браконьеров. Через определенный промежуток времени туман рассеивается, и лодка обнаруживается на расстоянии 12.2 км от катера. Затем лодка снова скрывается в тумане и уходит прямолинейно в неизвестном направлении. Известно, что скорость катера в 4.1 раза больше скорости браконьерской лодки

## 3.2 Код программы (Julia)

using DifferentialEquations  
using Plots  
  
n = 4.1  
s = 12.2  
fi = 3/4\*pi  
  
function f(r, p, t)  
 dr = r/sqrt(n^2-1)  
 return dr  
end  
  
function f2(t)  
 xt = tan(fi+pi)\*t  
 return xt  
end  
  
r0 = s/(n+1)  
  
theta0 = collect(LinRange(0, 2\*pi, 10000))  
prob = ODEProblem(f, r0, (0, 2\*pi))  
sol = solve(prob, saveat=theta0)  
  
t = collect(LinRange(0.00000001, 8, 1000))  
r1=[]  
theta1=[]  
for i in t  
 push!(r1, sqrt(i^2 + f2(i)^2))  
 push!(theta1, atan(f2(i)/i))  
end  
  
plot(sol, proj=:polar, label="катер")  
plot!(theta1, r1, proj=:polar, label="лодка")  
  
savefig("01jl.png")  
  
r0 = s/(n-1)  
  
theta0 = collect(LinRange(0, 2\*pi, 10000))  
prob = ODEProblem(f, r0, (0, 2\*pi))  
sol = solve(prob, saveat=theta0)  
  
t = collect(LinRange(0.00000001, 17, 1000))  
r1=[]  
theta1=[]  
for i in t  
 push!(r1, sqrt(i^2 + f2(i)^2))  
 push!(theta1, atan(f2(i)/i))  
end  
  
plot(sol, proj=:polar, label="катер")  
plot!(theta1, r1, proj=:polar, label="лодка")  
  
savefig("02jl.png")

## 3.3 Код программы (Python)

from math import \*  
import numpy as np  
from scipy.integrate import odeint  
import matplotlib.pyplot as plot  
  
n = 4.1  
s = 12.2  
fi = 3/4\*pi  
  
def f(theta, r):  
 dr = r/sqrt(n\*\*2 - 1)  
 return dr  
  
def f2(t):  
 xt = tan(fi+pi)\*t  
 return xt  
  
r0 = s/(n+1)  
  
theta = np.arange(0, 2\*pi, 0.01)  
r = odeint(f, r0, theta)  
  
t = np.arange(0.000000001, 20)  
r1 = np.sqrt(t\*\*2 + f2(t)\*\*2)  
theta1 = np.arctan(f2(t)/t)  
  
plot.rcParams["figure.figsize"] = (10, 10)  
  
plot.polar(theta, r, 'red', label = 'катер')  
plot.polar(theta1, r1, 'green', label = 'лодка')  
  
tmp = 0  
for i in range(len(theta)):  
 if round(theta[i], 2) == round(fi+pi, 2):  
 tmp = i  
print('Тета:', theta[tmp], "r:", r[tmp][0])  
  
plot.legend()  
plot.savefig("01.png",dpi=100)  
  
r0 = s/(n-1)  
  
theta = np.arange(0, 2\*pi, 0.01)  
r = odeint(f, r0, theta)  
  
t = np.arange(0.000000001, 20)  
r1 = np.sqrt(t\*\*2 + f2(t)\*\*2)  
theta1 = np.arctan(f2(t)/t)  
  
plot.rcParams["figure.figsize"] = (10, 10)  
  
plot.polar(theta, r, 'red', label = 'катер')  
plot.polar(theta1, r1, 'green', label = 'лодка')  
  
tmp = 0  
for i in range(len(theta)):  
 if round(theta[i], 2) == round(fi+pi, 2):  
 tmp = i  
print('Тета:', theta[tmp], "r:", r[tmp][0])  
  
plot.legend()  
plot.savefig("02.png",dpi=100)

## 3.4 Решение

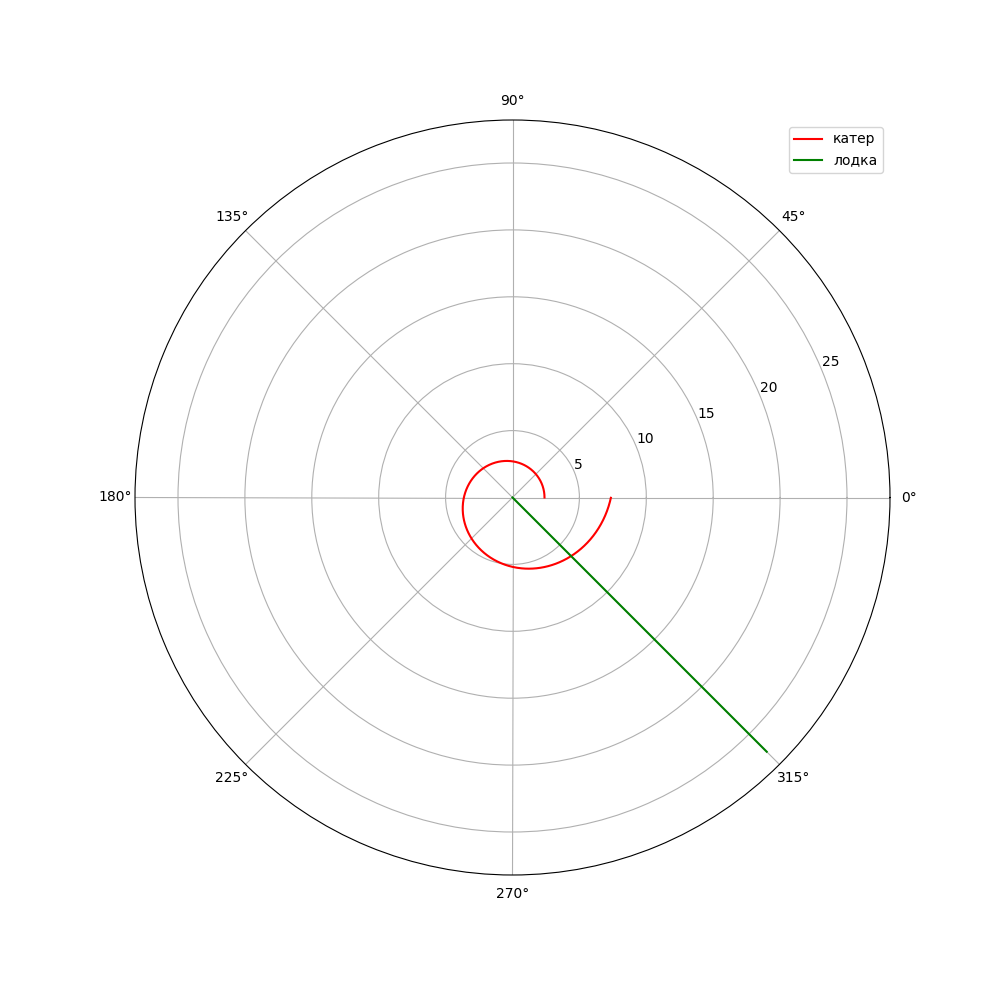


Figure 1: траектории для случая 1 (Python)

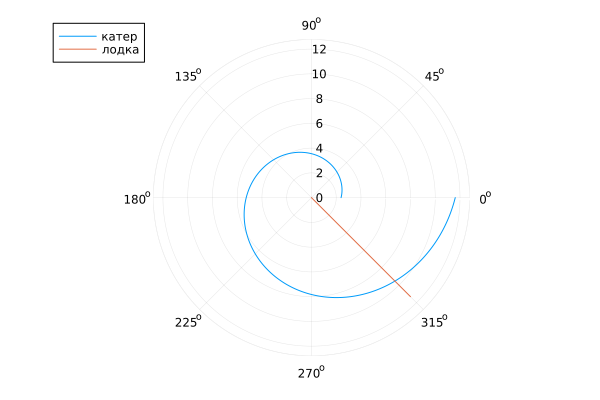


Figure 2: траектории для случая 1 (Julia)

Точка пересечения красного и зеленого графиков - точка пересечения катера и лодки, исходя из графика, имеет координаты

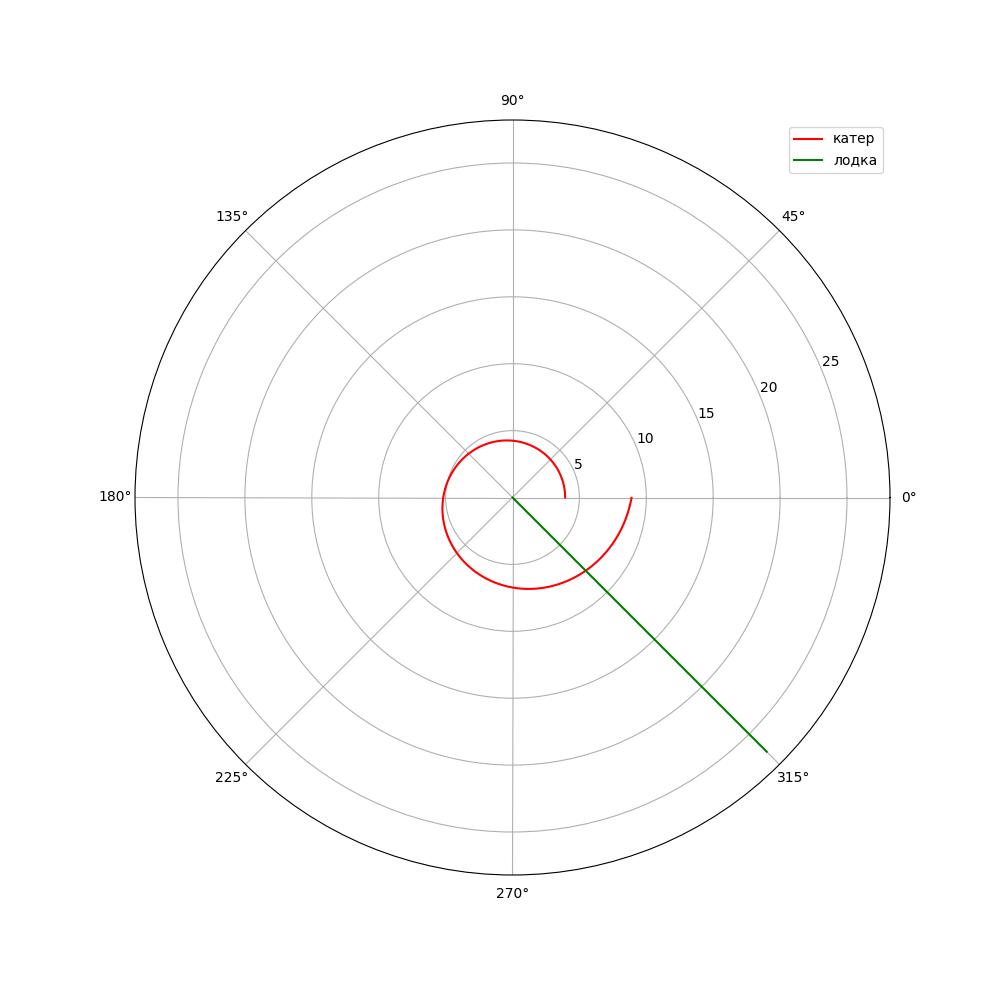


Figure 3: траектории для случая 2 (Python)

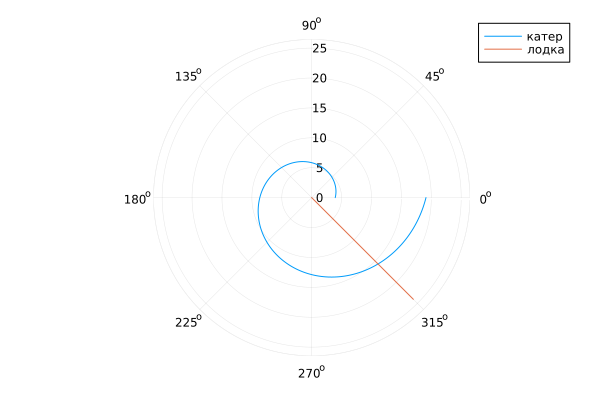


Figure 4: траектории для случая 2 (Julia)

Точка пересечения красного и зеленого графиков - точка пересечения катера и лодки, исходя из графика, имеет координаты

Наблюдаем, что при погоне «по часовой стрелке» для достижения цели потребуется пройти меньшее расстояние.

# 4 Выводы

Рассмотрели задачу о погоне. Провели анализ и вывод дифференциальных уравнений. Смоделировали ситуацию.