Лабораторная работа №2

Задача о погоне

Алиева Милена Арифовна

Содержание

# 1 Цель работы

Построить математическую модель для выбора правильной стратегии при решении примера задаче о погоне.

# 2 Задание

На море в тумане катер береговой охраны преследует лодку браконьеров. Через определенный промежуток времени туман рассеивается, и лодка обнаруживается на расстоянии 9,4 км от катера. Затем лодка снова скрывается в тумане и уходит прямолинейно в неизвестном направлении. Известно, что скорость катера в 3,7 раза больше скорости браконьерской лодки.

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. Формула для выбора варианта: (1132226430%70)+1 = 21 вариант.
2. Запишем уравнение описывающее движение катера, с начальными условиями для двух случаев (в зависимости от расположения катера относительно лодки в начальный момент времени).

Принимем 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 проходит через точку нахождения катера береговой охраны.

Траектория катера должна быть такой, чтобы и катер, и лодка все время были на одном расстоянии от полюса , только в этом случае траектория катера пересечется с траекторией лодки. Поэтому для начала катер береговой охраны должен двигаться некоторое время прямолинейно, пока не окажется на том же расстоянии от полюса, что и лодка браконьеров. После этого катер береговой охраны должен двигаться вокруг полюса удаляясь от него с той же скоростью, что и лодка браконьеров.

Чтобы найти расстояние (расстояние после которого катер начнет двигаться вокруг полюса), необходимо составить простое уравнение. Пусть через время катер и лодка окажутся на одном расстоянииx 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 (во втором случае ). Так как время одно и то же, то эти величины одинаковы. Тогда неизвестное расстояниеx можно найти из следующего уравнения:

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

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 , .

Получаем:

Из чего можно вывести:

Решение исходной задачи сводится к решению системы из двух дифференциальных уравнений:

С начальными условиями для первого случая:

Или для второго:

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

Начальные условия остаются прежними. Решив это уравнение, мы получим траекторию движения катера в полярных координатах.

1. Решить дифференциальное уравнение, расписанное в постановке задачи лабораторной работы, поможет библиотека DifferentialEquations. Итоговые изображения в полярных координатах будут строиться через библиотеку Plots. Загрузим дополнительно эти библиотеки.

Установим Julia, настроим работу в Jupyter notebook (рис. 1):

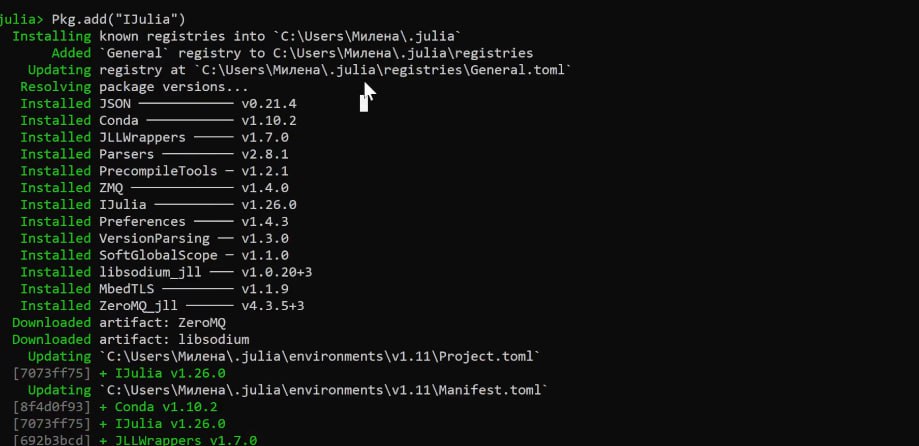


Рис. 1: Настройка Julia

1. Далее разработаем код для решения дифференциального уравнения и построения изображения:

Код:

using Plots  
using DifferentialEquations  
  
# расстояние от лодки до катера  
const a = 9.4  
const n = 3.7  
  
# расстояние начала спирали  
const r0 = a/(n + 1) # начальное расстояние для первого случая  
const r0\_2 = a/(n - 1) # начальное расстояние для второго случая  
  
# интервал времени  
const T = (0, 2\*pi) # интервал времени для первого случая  
const T\_2 = (-pi, pi) # интервал времени для второго случая  
  
# функция, описывающая изменение радиуса в зависимости от времени  
function F(u, p, t)  
 return u / sqrt(n\*n - 1) # уравнение ОДУ  
end  
  
# задача ОДУ для первого случая  
problem = ODEProblem(F, r0, T)  
  
# решение задачи ОДУ  
result = solve(problem, abstol=1e-8, reltol=1e-8)  
@show result.u # вывод значений радиуса  
@show result.t # вывод значений времени  
  
# генерация случайных индексов для выбора углов  
dxR = rand(1:size(result.t)[1])  
rAngles = [result.t[dxR] for i in 1:size(result.t)[1]] # выбор случайных углов  
  
# создание первого холста  
plt = plot(proj=:polar, aspect\_ratio=:equal, dpi = 1000, legend=true, bg=:white)  
  
# параметры для первого холста  
plot!(plt, xlabel="theta", ylabel="r(t)", title="Задача о погоне - случай 1", legend=:outerbottom)  
plot!(plt, [rAngles[1], rAngles[2]], [0.0, result.u[size(result.u)[1]]], label="Путь лодки", color=:blue, lw=1)  
scatter!(plt, rAngles, result.u, label="", mc=:blue, ms=0.0005) # точки пути лодки  
plot!(plt, result.t, result.u, xlabel="theta", ylabel="r(t)", label="Путь катера", color=:green, lw=1)  
scatter!(plt, result.t, result.u, label="", mc=:green, ms=0.0005) # точки пути катера  
  
# сохранение первого холста в файл  
savefig(plt, "lab02\_01.png")  
  
# задача ОДУ для второго случая  
problem = ODEProblem(F, r0\_2 , T\_2)  
result = solve(problem, abstol=1e-8, reltol=1e-8)  
dxR = rand(1:size(result.t)[1]) # генерация случайных индексов для второго случая  
rAngles = [result.t[dxR] for i in 1:size(result.t)[1]] # выбор случайных углов  
  
# создание второго холста  
plt1 = plot(proj=:polar, aspect\_ratio=:equal, dpi = 1000, legend=true, bg=:white)  
  
# параметры для второго холста  
plot!(plt1, xlabel="theta", ylabel="r(t)", title="Задача о погоне - случай 2", legend=:outerbottom)  
plot!(plt1, [rAngles[1], rAngles[2]], [0.0, result.u[size(result.u)[1]]], label="Путь лодки", color=:blue, lw=1)  
scatter!(plt1, rAngles, result.u, label="", mc=:blue, ms=0.0005) # точки пути лодки  
plot!(plt1, result.t, result.u, xlabel="theta", ylabel="r(t)", label="Путь катера", color=:green, lw=1)  
scatter!(plt1, result.t, result.u, label="", mc=:green, ms=0.0005) # точки пути катера  
  
# сохранение второго холста в файл  
savefig(plt1, "lab02\_02.png")

1. Получим два изображения - для первого и для второго случаев (рис. 2), (рис. 3):



Рис. 2: Изображение для первого случая



Рис. 3: Изображение для второго случая

# 4 Выводы

В процессе выполнения лабораторной работы №2 я научилась строить математическую модель для выбора правильной стратегии при решении примера задаче о погоне.