Отчёт по лабораторной работе №2

НКНбд-01-21

Подлесный Иван Сергеевич

Содержание

1 Введение

Математи́ческая моде́ль — математическое представление реальности[1], один из вариантов модели как системы, исследование которой позволяет получать информацию о некоторой другой системе. Математическая модель, в частности, предназначена для прогнозирования поведения реального объекта, но всегда представляет собой ту или иную степень его идеализации.

Математи́ческим моделированием называют как саму деятельность, так и совокупность принятых приёмов и техник построения и изучения математических моделей.

2 Цель работы

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

Для данной задачи: - На море в тумане катер береговой охраны преследует лодку браконьеров. Через определенный промежуток времени туман рассеивается, и лодка обнаруживается на расстоянии 11,5 км от катера. Затем лодка снова скрывается в тумане и уходит прямолинейно в неизвестном направлении. Известно, что скорость катера в 3,5 раза больше скорости браконьерской лодки

Вариант вычислялся по формуле номер ст.билета % кол-во заданий. Результатом стало число n % 70 = 32

3 Ход работы

1. Установили Julia, используя пакет apt bash sudo apt-get install julia, и внутренний пакет Plots, используя команду julia using Pkg; Pkg.add("Plots").

2. Вычислили расстояние между лодкой (браконьеров) и катером (охрана), используя формулу $\frac{x}{\nu} = \frac{s \pm x}{k * \nu}$, где s = начальное расстояние между лодкой и катером равный 14.4 км и k = коэффициент во сколько раз скорость катера выше чем скорость лодки. В итоге получили значения $x_1 = \frac{11.5}{4.5}$ и $x_2 = \frac{11.5}{2.5}$

3. Полагая, что 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 ν. Для этого скорость катера раскладываем на две составляющие: ν_r - радиальная скорость и ν_t - тангенциальная скорость. $\nu_r = \frac{dr}{dt}$. Нам нужно, чтобы эта скорость была равна скорости лодки, поэтому полагаем $\frac{dr}{dt} = \nu$. Тангенциальная скорость – это линейная скорость вращения катера относительно полюса. Она равна произведению угловой скорости $\frac{d\theta}{dt}$ на радиус, то есть $\nu_t = r\frac{d\theta}{dt}$. Отсюда, используя теорему Пифагора находим ν_t, которая равна $\sqrt{(k * \nu)^2 - \nu^2}$. В данном варианте получаем $\nu_t = \sqrt{11.25}\nu$.

4. Решение исходной задачи сводится к решению системы из двух дифференциальных уравнений $\begin{cases} \frac{d\nu}{dt} = \nu \\ r\frac{d\theta}{dt} = \sqrt{11.25}\nu \end{cases}$. После интегрирования

получаем r = $Ce^{\frac{\theta}{\sqrt{11.25}}}$

5. Переписываем все в julia и получаем

```
using Plots


const s = 11.5

const k = 3.5

const ThetaCrdeg = 320

const dTheta = 0.01

const MaxTheta = 4*pi

const cases =["F","S"]


function F(theta)

return r0 * exp.(theta / sqrt.(k^2 - 1))

end


for case in cases

global r0 = -1

theta0 = -1


if case=="F"

r0 = s / (k + 1)

theth0 = 0

else

r0 = s / (k - 1)

theta0 = -pi

end


theta1 = theta0 + MaxTheta
```

```
thetaCop = theta0 : dTheta : theta1
thetaCrook = ThetaCrdeg * pi / 180 + 2 *theta0

plt = plot( proj=:polar, aspect_ratio=:equal, dpi=500, title="Lab-#2" * case * "Case", legend=true )
plot!(plt, [theta0, theta0], [s, F(theta0)], label=false, color=:red)
plot!(plt, thetaCop, F, label=:"Cop's path" , color=:red)
plot!(plt, [0, thetaCrook],[0, F(thetaCrook)+20], label=:"Crook trajectory", color=:green)

plot!(plt, [theta0], [s], seriestype=:scatter, label=:"Cop's starting position", color=:red)
plot!(plt, [0], [0], seriestype=:scatter, label=:"Crook starting position", color=:green)
plot!(plt, [thetaCrook], [F(thetaCrook)], seriestype=:scatter, label=:"intersection point", color=:yellow)

savefig(plt, "Lab-#2" * case * "Case#.png")
display(plt)
end
```

6. Результат случая s + x (рис. 3.1)

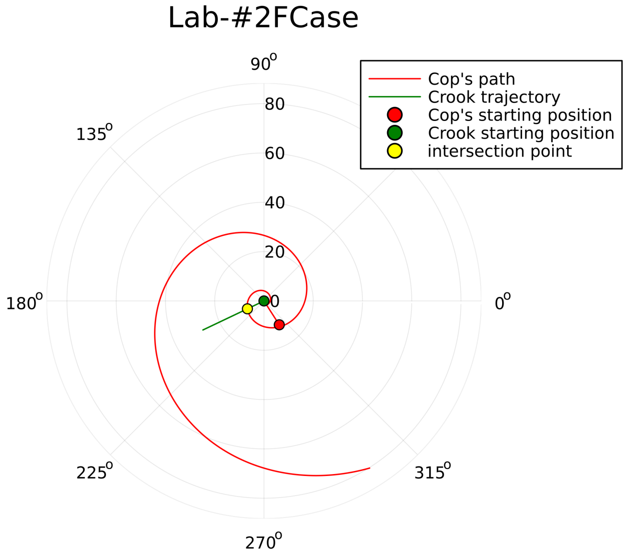

Рис. 3.1: Результат при случае s + x

7. Результат случая s - x (рис. 3.2)

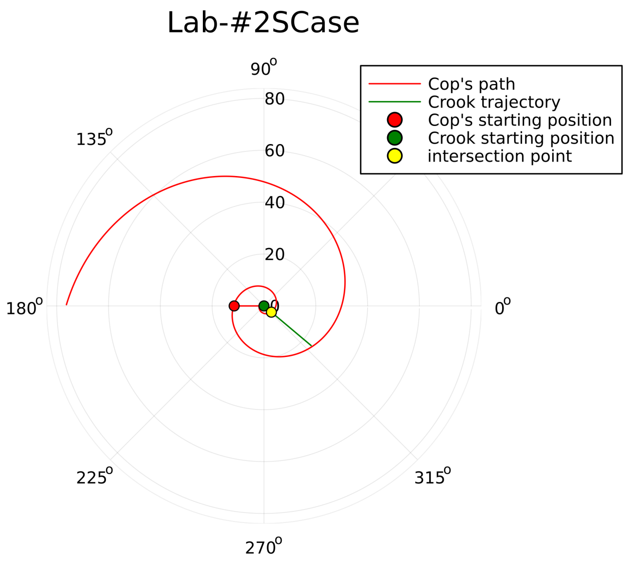

Рис. 3.2: Результат при случае s - x

4 Вывод

Во время выполнения лабораторной работы, мы получили базовые знания работы с julia и математическим моделированием.