Лабораторная работа 2

Задача о погоне

Фаик Карим

Содержание

# Цель работы

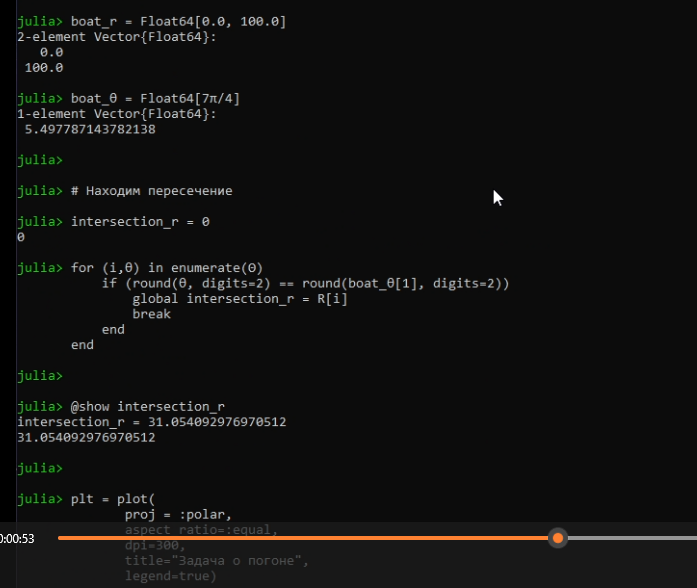
Решить задачу о погоне, на примере лодками с браконьером и береговой охраной.

# Задание

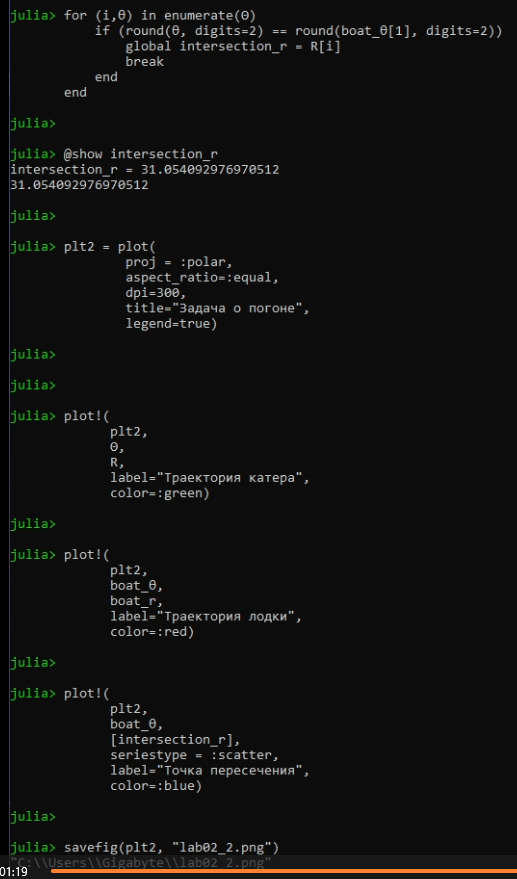
1. Провести аналогичные рассуждения и вывод дифференциальных уравнений, если скорость катера больше скорости лодки в n раз (значение n задайте самостоятельно)
2. Построить траекторию движения катера и лодки для двух случаев. (Задайте самостоятельно начальные значения)
3. Определить по графику точку пересечения катера и лодки.

# Выполнение лабораторной работы

1. Зададим начальные условия (вариант 62)
2. Будем вести отсчет в полярных координатах. Полюс у нас это место обнаружения браконьеров.
3. Чтобы найти расстояние x(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 (во втором случае ). Так как время одно и то же, то эти величины одинаковы. Составим уравнения и найдем растояние (@fig:001)



Нахождение первого решения



Нахождение второго решения

1. 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 v. Для этого скорость катера раскладываем на радиальную [@rad:math] и тангенциальную [@tan:math] скорости
2. Давайте просчитаем траекторию движения браконьеров и движения лодки охраны и найдем точки пересечения при помощи следующего кода

# Код

```julia

using Plots using DifferentialEquations

function F(du,u,p,t) r, θ = u du[1] = 2 du[2] = sqrt(19.25) / u[1] end

r₀ = 14.4/5.7 h = 0.1 θ₀ = 0.0 tspan = (0, 100) prob = ODEProblem(F, [r₀, θ₀], tspan) sol = solve(prob, dtmax=h)

#Достаем значения R = [u[1] for u in sol.u] Θ = [u[2] for u in sol.u]

boat\_r = Float64[0.0, 100.0] boat\_θ = Float64[7π/4]

# Находим пересечение

intersection\_r = 0 for (i,θ) in enumerate(Θ) if (round(θ, digits=2) == round(boat\_θ[1], digits=2)) global intersection\_r = R[i] break end end

@show intersection\_r

plt = plot( proj = :polar, aspect\_ratio=:equal, dpi=300, title=“Задача о погоне”, legend=true)

plot!( plt, Θ, R, label=“Траектория катера”, color=:green)

plot!( plt, boat\_θ, boat\_r, label=“Траектория лодки”, color=:red)

plot!( plt, boat\_θ, [intersection\_r], seriestype = :scatter, label=“Точка пересечения”, color=:blue)

savefig(plt, “lab02\_1.png”)

r₀ = 18.1/3.7 θ₀ = π

prob = ODEProblem(F, [r₀, θ₀], tspan) sol = solve(prob, dtmax=h)

#Достаем значения R = [u[1] for u in sol.u] Θ = [u[2] for u in sol.u]

boat\_r = Float64[0.0, 100.0]

# Находим пересечение

for (i,θ) in enumerate(Θ) if (round(θ, digits=2) == round(boat\_θ[1], digits=2)) global intersection\_r = R[i] break end end

@show intersection\_r

plt2 = plot( proj = :polar, aspect\_ratio=:equal, dpi=300, title=“Задача о погоне”, legend=true)

plot!( plt2, Θ, R, label=“Траектория катера”, color=:green)

plot!( plt2, boat\_θ, boat\_r, label=“Траектория лодки”, color=:red)

plot!( plt2, boat\_θ, [intersection\_r], seriestype = :scatter, label=“Точка пересечения”, color=:blue)

savefig(plt2, “lab02\_2.png”)

1. Далее получим следующие траектории движения



Траектория 1



Траектория 2

# Выводы

По мере выполнения данной работы я научился работать с языком Julia посредством решения задачи о погоне на примере лодки с браконьерами и береговой охраны