

# Модель гармонических колебаний

Лабораторная работа №4

---

Покрас Илья Михайлович

2 марта 2023

Российский университет дружбы народов, Москва, Россия

Целью данной лабораторной работы является построение математических моделей гармонических колебаний.

Построить фазовый портрет гармонического осциллятора и решить уравнения гармонического осциллятора для следующих случаев: 1. Колебания гармонического осциллятора без затуханий и без действий внешней силы 2. Колебания гармонического осциллятора с затуханием и без действий внешней силы 3. Колебания гармонического осциллятора с затуханием и под действием внешней силы

## Код Julia:

```
using Plots
using DifferentialEquations

w1 = 7.7
g1 = 0.0
w2 = 7.7
g2 = 7.0
w3 = 7.7
g3 = 7.0
X0 = 0.7
Y0 = -0.7

function ode_fn1(du, u, p, t)
    du[1] = u[2]
    du[2] = -w1*u[1] - g1*u[2]
end

function ode_fn2(du, u, p, t)
    du[1] = u[2]
    du[2] = -w2*u[1] - g2*u[2]
end

function ode_fn3(du, u, p, t)
    du[1] = u[2]
    du[2] = -w3*u[1] - g3*u[2] + 0.7*sin(7*t)
end

tspan = (0.0, 77.0)
prob1 = ODEProblem(ode_fn1, [X0, Y0], tspan)
sol1 = solve(prob1, dtmax=0.05)
prob2 = ODEProblem(ode_fn2, [X0, Y0], tspan)
sol2 = solve(prob2, dtmax=0.05)
prob3 = ODEProblem(ode_fn3, [X0, Y0], tspan)
sol3 = solve(prob3, dtmax=0.05)

X1 = [u[1] for u in sol1.u]
Y1 = [u[2] for u in sol1.u]
X2 = [u[1] for u in sol2.u]
Y2 = [u[2] for u in sol2.u]
X3 = [u[1] for u in sol3.u]
Y3 = [u[2] for u in sol3.u]
T = [t for t in sol1.t]

plt1 = plot(dpi=300, title="Фазовый портрет", legend=false)
plot!(plt1, X1, Y1, color=:red)

plt2 = plot(dpi=300, title="Решение уравнения", legend=false)
plot!(plt2, T, X1, color=:green)
plot!(plt2, T, Y1, color=:blue)

plt3 = plot(dpi=300, title="Фазовый портрет", legend=false)
plot!(plt3, X2, Y2, color=:red)

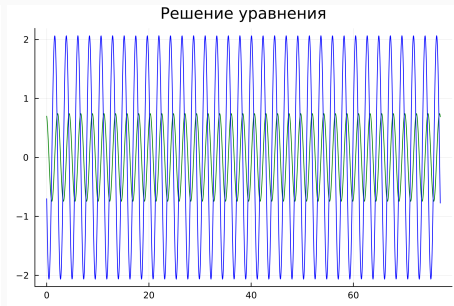
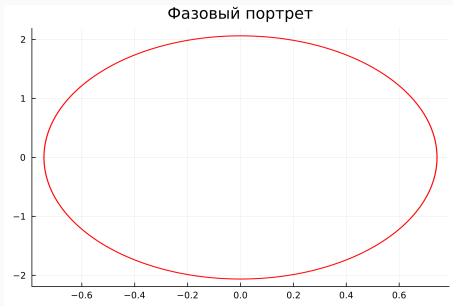
plt4 = plot(dpi=300, title="Решение уравнения", legend=false)
plot!(plt4, T, X2, color=:green)
plot!(plt4, T, Y2, color=:blue)

plt5 = plot(dpi=300, title="Фазовый портрет", legend=false)
plot!(plt5, X3, Y3, color=:red)

plt6 = plot(dpi=300, title="Решение уравнения", legend=false)
plot!(plt6, T, X3, color=:green)
plot!(plt6, T, Y3, color=:blue)

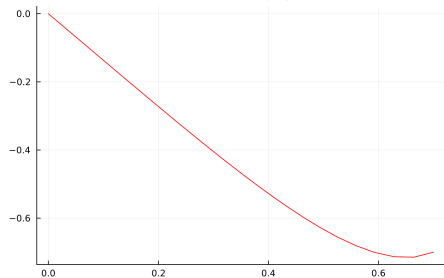
savefig(plt1, "model1_1.png")
savefig(plt2, "model1_2.png")
savefig(plt3, "model2_1.png")
savefig(plt4, "model2_2.png")
savefig(plt5, "model3_1.png")
savefig(plt6, "model3_2.png")
```

Первый случай:

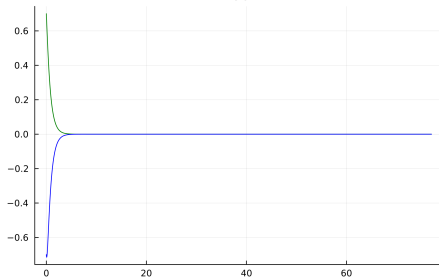


Второй случай:

Фазовый портрет

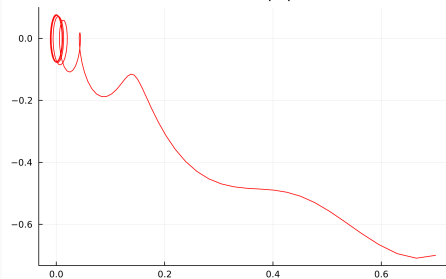


Решение уравнения

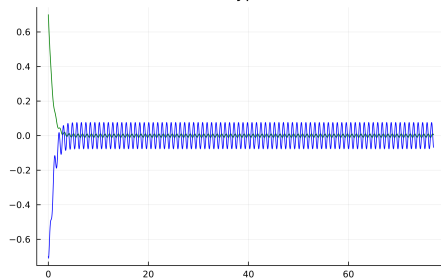


Третий случай:

Фазовый портрет



Решение уравнения



Код OpenModelica:

```
model Model1
  Real x(start = 0.7);
  Real y(start = -0.7);
  Real w = 7.7;
  Real g = 0.0;
  Real t = time;
equation
  der(x) = y;
  der(y) = -w*x - g*y;
  annotation(experiment(StartTime = 0, StopTime = 77, Tolerance = 1e-6, Interval = 0.05));
end Model1;
```

Рис. 1: Код OpenModelica для первого случая



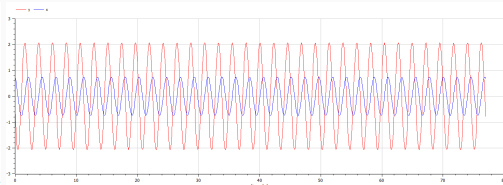
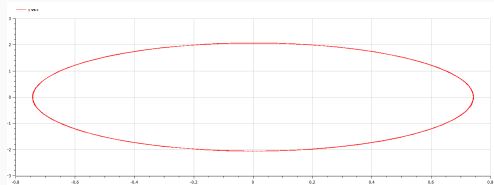
```
model Model2
  Real x(start = 0.7);
  Real y(start = -0.7);
  Real w = 7.7;
  Real g = 7.0;
  Real t = time;
equation
  der(x) = y;
  der(y) = -w*x - g*y;
  annotation(experiment(StartTime = 0, StopTime = 77, Tolerance = 1e-6, Interval = 0.05));
end Model2;
```

Рис. 2: Код OpenModelica для второго случая

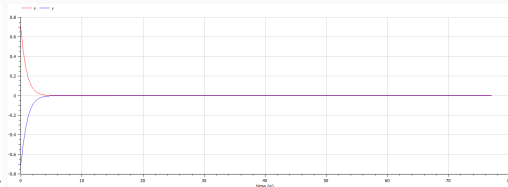
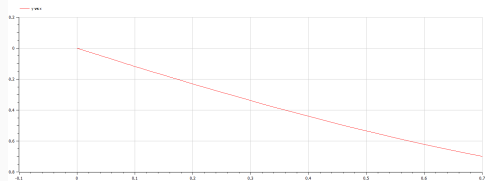
```
model Model2
  Real x(start = 0.7);
  Real y(start = -0.7);
  Real w = 7.7;
  Real g = 7.0;
  Real t = time;
equation
  der(x) = y;
  der(y) = -w*x - g*y;
  annotation(experiment(StartTime = 0, StopTime = 77, Tolerance = 1e-6, Interval = 0.05));
end Model2;
```

Рис. 3: Код OpenModelica для третьего случая

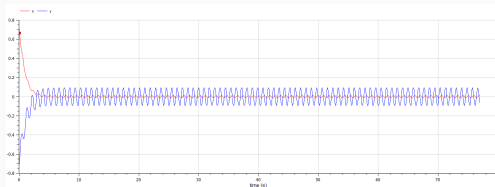
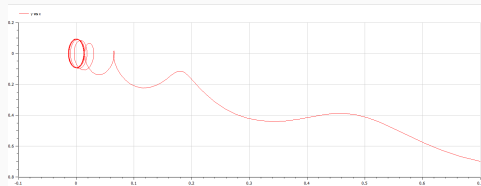
Первый случай:



Второй случай:



Третий случай:



В результате проделанной работы был написан код на Julia и OpenModelica для решения 3 случаев движения гармонического осциллятора.