презентаця лабораторной работе №4

Модель гармонических колебаний

Габриэль Тьерри

Содержание

# Цель работы

* Научиться решать линейное однородное дифференциальное уравнение второго порядка.
* Построить модель линейного гармонического осциллятора без затухания/ с затуханием/ с действием внешней силы.
* Построить фазовые портреты всех моделей.
* Отработать навыки решения систем дифференциальных уравнений на языке Julia, Openmodelica

# Задание

Постройте фазовый портрет гармонического осциллятора и решение уравнения гармонического осциллятора для следующих случаев

1. Колебания гармонического осциллятора без затуханий и без действий внешней силы
2. Колебания гармонического осциллятора c затуханием и без действий внешней силы
3. Колебания гармонического осциллятора c затуханием и под действием внешней силы
4. На интервале
5. (шаг 0.05) с начальными условиями x0 = 2, y0 = -0.2

# Теоретическое введение

**Гармонический осциллятор** - система, которая при выведении ее из положения равновесия испытывает действие возращающей силы , пропорциональной смещению :

Если — единственная сила, действующая на систему, то систему называют **простым** или **консервативным гармоническим осциллятором**. Свободные колебания такой системы представляют собой периодическое движение около положения равновесия (гармонические колебания). Частота и амплитуда при этом постоянны, причём частота не зависит от амплитуды.

Если имеется ещё и сила трения (затухание), пропорциональная скорости движения (вязкое трение), то такую систему называют **затухающим** или **диссипативным осциллятором**. Если трение не слишком велико, то система совершает почти периодическое движение — синусоидальные колебания с постоянной частотой и экспоненциально убывающей амплитудой. Частота свободных колебаний затухающего осциллятора оказывается несколько ниже, чем у аналогичного осциллятора без трения.

Если осциллятор предоставлен сам себе, то говорят, что он совершает **свободные колебания**. Если же присутствует внешняя сила (зависящая от времени), то говорят, что осциллятор испытывает **вынужденные колебания**.

Механическими примерами гармонического осциллятора являются математический маятник (с малыми углами отклонения), груз на пружине, торсионный маятник и акустические системы. Среди немеханических аналогов гармонического осциллятора можно выделить электрический гармонический осциллятор. @Oscilliator

Для *консервативного гармонического осциллятора*, используя второй закон Ньютона, имеем @Lab:

Обозначим и подставим

Для *затухающего гармонического осциллятора*, используя второй закон Ньютона, имеем @Lab:

Обозначим , и подставим

Для *вынужденных колебаний*, используя второй закон Ньютона, имеем @Oscilliator:

Обозначим , и подставим

Для решения данных дифференциальных уравнений будем составлять систему, в которой обозначим за , будем оставлять без изменений.

Например, для затухающих гармонических колебаний имеем:

**Фазовая плоскость** — координатная плоскость, в которой по осям координат откладываются какие-либо две переменные (фазовые координаты), однозначно определяющие состояние системы второго порядка.

Каждая точка фазовой плоскости отражает одно некоторое состояние системы и называется *фазовой, изображающей или представляющей* точкой

Изменение состояния системы отображается на фазовой плоскости движением этой точки. След от движения изображающей точки называется **фазовой траекторией**.

Полная совокупность всевозможных различных фазовых траекторий — это **фазовый портрет**.

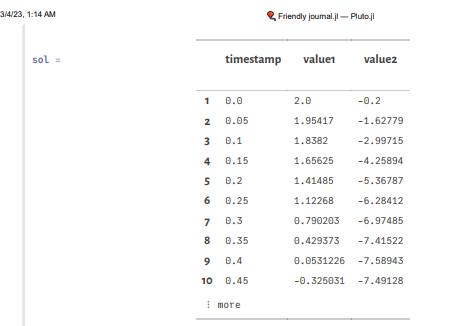
В нашей задаче на оси абсцисс фазовой плоскости откладываются значения параметра , например, величина отклонения от равновесия; на оси ординат откладываются значения первой производной по времени() — скорости перемещения. @Phase

# Выполнение лабораторной работы

1. Построим решение уравнения гармонического осциллятора без затухания. В нашем случае оно имеет вид
2. Для того, чтобы решить данное уравнение перепишем его в виде системы из двух дифференциальных уравнений первого порядка:
3. Далее запишем начальные условия, которые в нашем случае имеют вид:

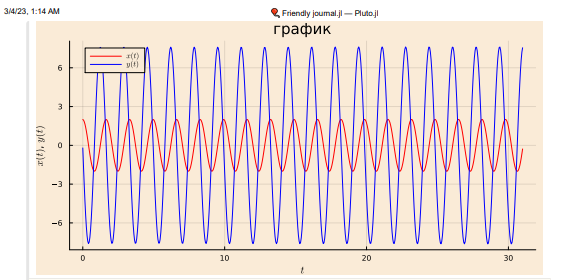
Теперь используем язык программирования Julia для получения численного решения и построения фазового портрета

begin  
 import Pkg  
 Pkg.add("LaTeXStrings")  
 Pkg.activate()  
 using DifferentialEquations  
 using LaTeXStrings  
 import Plots  
end  
  
begin  
 X0 = 2.0  
 Y0 = -0.2  
 w = 14.4  
 t0 = 0.0  
 tmax = 31.0  
 tspan = (t0, tmax)  
end  
  
function F!(du, u, p, t)  
 du[1] = u[2]  
 du[2] = -w\*u[1]  
end  
  
begin  
 U0 = [X0, Y0]  
 T = [t0, tmax]  
 prob = ODEProblem(F!, U0, T)  
end



sol

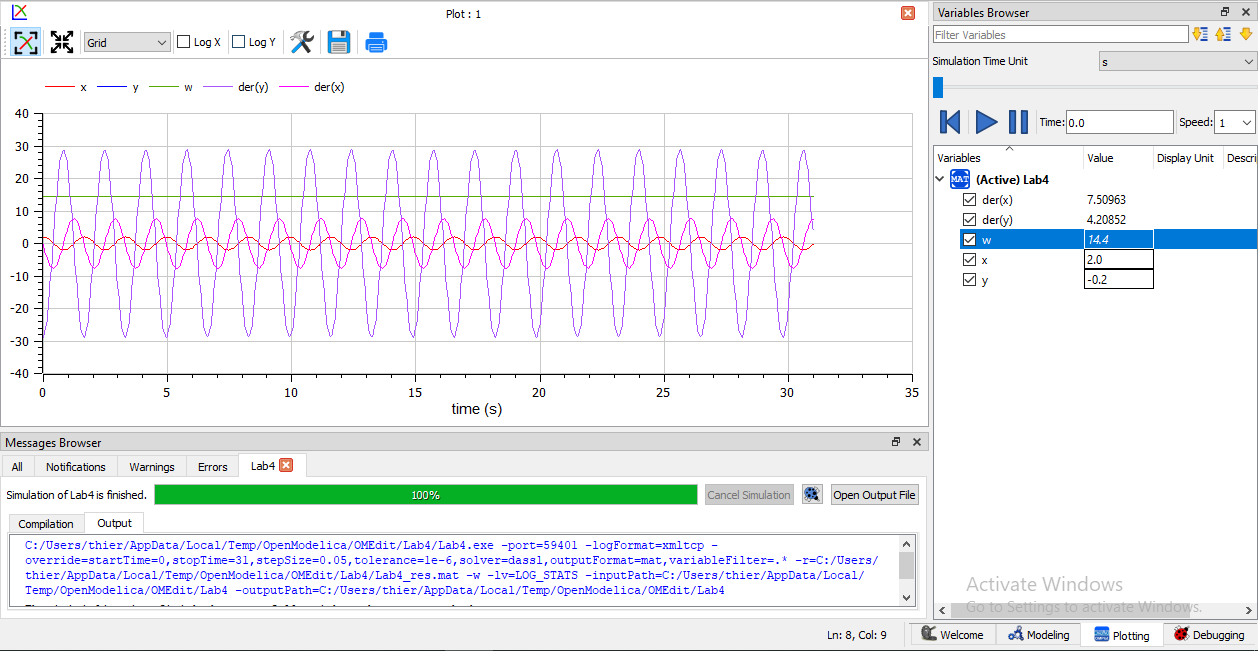
sol = solve(prob, saveat = 0.05)  
  
begin  
 Time = sol.t  
 const X = Float64[]  
 const Y = Float64[]  
 for u in sol.u  
 x, y = u  
 push!(X, x)  
 push!(Y, y)  
 end  
 X,Y  
end  
  
begin  
 #график  
 fig = Plots.plot(  
 layout = (1, 1),  
 dpi = 150,  
 grid =:xy,  
 gridcolor =:black,  
 gridwidth = 1,  
 background\_color=:antiquewhite,  
 # aspect\_ratio=: equal,  
 size = (800, 400),  
 plot\_title="график",  
 )  
 Plots.plot!(  
 fig[1],  
 Time,  
 [X Y],  
 xlabel = L"$t$",  
 ylabel = L"$x(t)$, $y(t)$",  
 color =[ :red :blue ],  
 label = [L"$x(t)$" L"$y(t)$"]  
 )  
end



Фазовый портрет №1(Julia)

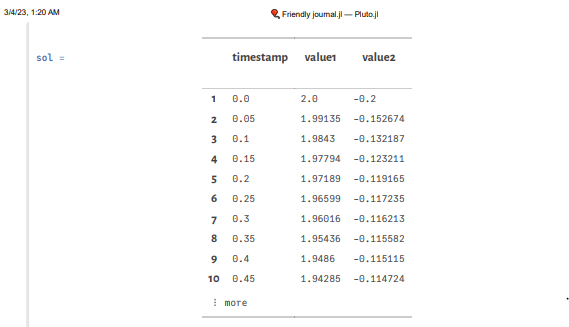
проделаем те же самые действия в Openmodelica

model Lab4  
Real x(start=2);  
Real y(start=-0.2);  
parameter Real w =14.4;  
equation  
der(x) = y;  
der(y) = -w\*x;  
end Lab4;

Получим фазовый портрет 

1. Построим решение уравнения для колебаний гармонического осциллятора с затуханием и без действий внешней силы. В нашем случае оно имеет вид
2. Для того, чтобы решить данное уравнение перепишем его в виде системы из двух дифференциальных уравнений первого порядка:
3. Далее запишем начальные условия, которые в нашем случае имеют вид:
4. Теперь используем язык программирования Julia для получения численного решения и построения фазового портрета

begin  
 import Pkg  
 Pkg.add("LaTeXStrings")  
 Pkg.activate()  
 using DifferentialEquations  
 using LaTeXStrings  
 import Plots  
end  
  
begin  
 X0 = 2.0  
 Y0 = -0.2  
 g = 17.0  
 w = 1.0  
 t0 = 0.0  
 tmax = 31.0  
 tspan = (t0, tmax)  
end  
  
function F!(du, u, p, t)  
 du[1] = u[2]  
 du[2] = -g\*u[2]-w\*u[1]  
end  
  
begin  
 U0 = [X0, Y0]  
 T = [t0, tmax]  
 prob = ODEProblem(F!, U0, T)  
end



Sol

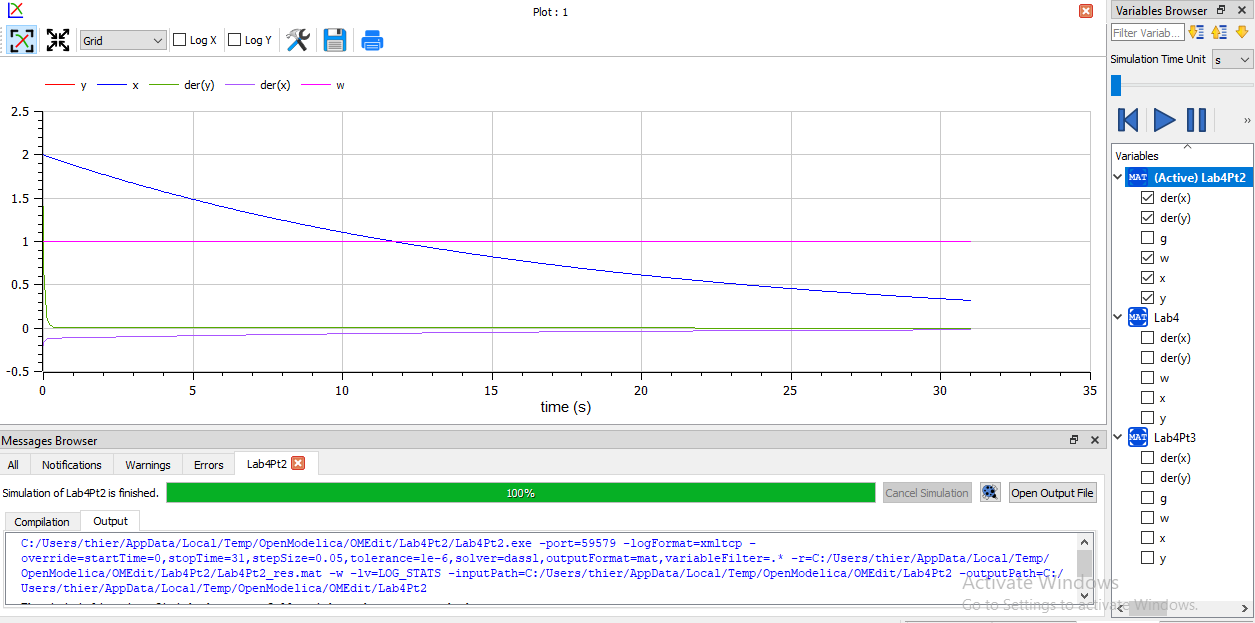
sol = solve(prob, saveat = 0.05)  
  
begin  
 Time = sol.t  
 const X = Float64[]  
 const Y = Float64[]  
 for u in sol.u  
 x, y = u  
 push!(X, x)  
 push!(Y, y)  
 end  
 X,Y  
end  
  
begin  
 #график  
 fig = Plots.plot(  
 layout = (1, 1),  
 dpi = 150,  
 grid =:xy,  
 gridcolor =:black,  
 gridwidth = 1,  
 background\_color=:antiquewhite,  
 # aspect\_ratio=: equal,  
 size = (800, 400),  
 plot\_title="график",  
 )  
 Plots.plot!(  
 fig[1],  
 Time,  
 [X Y],  
 xlabel = L"$t$",  
 ylabel = L"$x(t)$, $y(t)$",  
 color =[ :red :blue ],  
 label = [L"$x(t)$" L"$y(t)$"]  
 )  
end



Фазовый портрет №2(Julia)

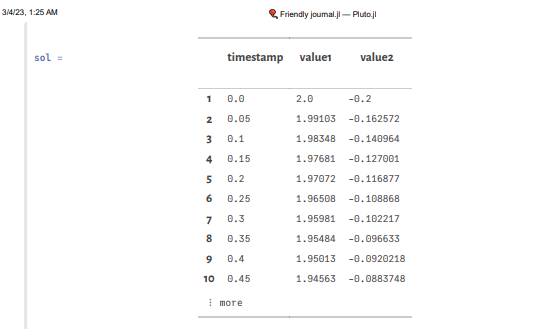
проделаем те же самые действия в Openmodelica

model Lab4Pt2  
Real x(start=2);  
Real y(start=-0.2);  
parameter Real g = 17;  
parameter Real w =1;  
equation  
der(x) = y;  
der(y) = -g\*y-w\*x;  
end Lab4Pt2;

Получим фазовый портрет 

1. Построим решение уравнения для колебаний гармонического осциллятора с затуханием и под действием внешней силы. В нашем случае оно имеет вид
2. Для того, чтобы решить данное уравнение перепишем его в виде системы из двух дифференциальных уравнений первого порядка:
3. Далее запишем начальные условия, которые в нашем случае имеют вид:
4. Теперь используем язык программирования Julia для получения численного решения и построения фазового портрета

begin  
 import Pkg  
 Pkg.add("LaTeXStrings")  
 Pkg.activate()  
 using DifferentialEquations  
 using LaTeXStrings  
 import Plots  
end  
  
  
begin  
 X0 = 2.0  
 Y0 = -0.2  
 g = 15.0  
 w = 1.0  
 t0 = 0.0  
 tmax = 31.0  
 tspan = (t0, tmax)  
end  
  
function P(t)  
 return 0.7\*sin(3\*t)  
end  
  
function F!(du, u, p, t)  
 du[1] = u[2]  
 du[2] = -g\*u[2]-w\*u[1] + P(t)  
end  
  
begin  
 U0 = [X0, Y0]  
 T = [t0, tmax]  
 prob = ODEProblem(F!, U0, T)  
end



Sol

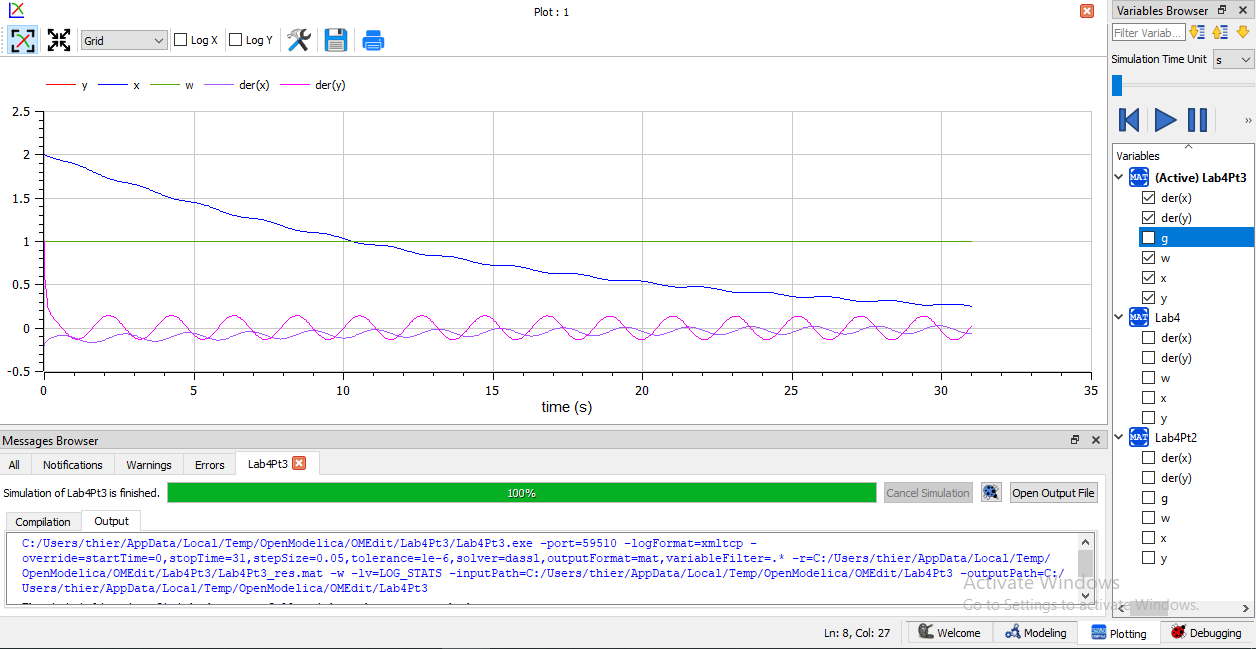
sol = solve(prob, saveat = 0.05)  
  
begin  
 Time = sol.t  
 const X = Float64[]  
 const Y = Float64[]  
 for u in sol.u  
 x, y = u  
 push!(X, x)  
 push!(Y, y)  
 end  
 X,Y  
end  
  
begin  
 #график  
 fig = Plots.plot(  
 layout = (1, 1),  
 dpi = 150,  
 grid =:xy,  
 gridcolor =:black,  
 gridwidth = 1,  
 background\_color=:antiquewhite,  
 # aspect\_ratio=: equal,  
 size = (800, 400),  
 plot\_title="график",  
 )  
 Plots.plot!(  
 fig[1],  
 Time,  
 [X Y],  
 xlabel = L"$t$",  
 ylabel = L"$x(t)$, $y(t)$",  
 color =[ :red :blue ],  
 label = [L"$x(t)$" L"$y(t)$"]  
 )  
end



Фазовый портрет №3(Julia)

проделаем те же самые действия в Openmodelica

model Lab4Pt3  
Real x(start=2);  
Real y(start=-0.2);  
parameter Real g = 15;  
parameter Real w = 1;  
equation  
der(x) = y;  
der(y) = -g\*y-w\*x + 0.7\*sin(3\*time);  
end Lab4Pt3;

Получим фазовый портрет 

# Ответы на вопросы

1. Простейшая модель гармонических колебаний:
2. Осциллятор - система, совершающая колебания, то есть показатели которой периодически повторяются во времени.
3. Модель математического маятника:
4. Для перехода от дифференциального уравнения второго порядка к системе дифференциальных уравнений первого порядка, необходимо в уравнении заменить все производные первого порядка на новую переменную, а также записать дополнительное уравнения, в котором будет обозначено, что первая производная равна новой переменной
5. **Фазовый портрет** — это геометрическое представление траекторий динамической системы на фазовой плоскости
6. **Фазовая траектория** - кривая в фазовом пространстве, составленная из точек, представляющих состояние динамической системы в последовательные моменты времени в течение всего времени эволюции.

# Выводы

* Научились решать линейное однородное дифференциальное уравнение второго порядка.
* Построиили модель линейного гармонического осциллятора без затухания/ с затуханием/ с действием внешней силы.
* Построили фазовые портреты всех моделей. Увидели, что при реализации на Julia и Openmodelica портреты совпадают.
* Отработали навыки решения систем дифференциальных уравнений на языке Julia, Openmodelica

# Список литературы