

Лабораторная работа № 3

Модель гармонический колебаний

Покрас Илья Михайлович

Содержание

Цели работы	4
Задание	5
Теоретическое введение:	6
Выполнение лабораторной работы	7
Код на Julia:	7
Код OpenModelica:	16
Вывод	21
Список Литературы	22

Список иллюстраций

1	Переменные и библиотеки	7
2	ОДУ	8
3	Решение ОДУ	8
4	Копирование данных через for	9
5	Визуализация	10
6	Фазовый портрет первого случая	11
7	Решение первого случая	12
8	Фазовый портрет второго случая	13
9	Решение второго случая	14
10	Фазовый портрет третьего случая	15
11	Решение третьего случая	16
12	Код OpenModelica для первого случая	17
13	Код OpenModelica для второго случая	17
14	Код OpenModelica для третьего случая	17
15	Фазовый портрет первого случая	18
16	Решение первого случая	18
17	Фазовый портрет второго случая	19
18	Решение второго случая	19
19	Фазовый портрет третьего случая	20
20	Решение третьего случая	20

Цели работы

Целью данной лабораторной работы является построение математической модели гармонических колебаний.

Задание

Построим фазовый портрет гармонического осциллятора и решение уравнения гармонического осциллятора для следующих случаев:

1. Колебания гармонического осциллятора без затуханий и без действий внешней сил: $\ddot{x} + 7.7x = 0$
2. Колебания гармонического осциллятора с затуханием и без действий внешней силы: $\ddot{x} + 7\dot{x} + 7.7x = 0$
3. Колебания гармонического осциллятора с затуханием и под действием внешней силы: $\ddot{x} + 7\dot{x} + 7.7x = 0.7\sin(7t)$

Теоретическое введение:

Движение груза на пружинке, маятника, заряда в электрическом контуре, а также эволюция во времени многих систем в физике, химии, биологии и других науках при определенных предположениях можно описать одним и тем же дифференциальным уравнением, которое в теории колебаний выступает в качестве основной модели. Эта модель называется линейным гармоническим осциллятором. Уравнение свободных колебаний гармонического осциллятора имеет следующий вид:

$$\ddot{x} + 2\gamma\dot{x} + \omega_0^2 x = 0$$

где x – переменная, описывающая состояние системы (смещение груза, заряд конденсатора и т.д.), γ – параметр, характеризующий потери энергии (трение в механической системе, сопротивление в контуре), ω_0 – собственная частота колебаний, t – время.

Выполнение лабораторной работы

Код на Julia:

Инициализируем библиотеки для дальнейшей работы. Далее создадим переменные для начальных X и Y и опишем ω и γ для 3 случаев (@fig:001).

```
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
```

Рис. 1: Переменные и библиотеки

Создаим с помощью Differential Equations системы уравнений для 3 случаев(@fig:002).

```

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

```

Рис. 2: ОДУ

С помощью solve получим решения ОДУ для всех случаев(@fig:003).

```

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

```

Рис. 3: Решение ОДУ

Сохраним данные решений в отдельные вектора для каждого случая (@fig:004).


```
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]
```

Рис. 4: Копирование данных через for

Визуализируем решение с помощью Plots(@fig:005)

```

plt1 = plot(dpi=300, title="Фазовый портрет", legend=false)
plot!(plt1,  $X_1$ ,  $Y_1$ , color=:red)

plt2 = plot(dpi=300, title="Решение уравнения", legend=false)
plot!(plt2,  $T$ ,  $X_1$ , color=:green)
plot!(plt2,  $T$ ,  $Y_1$ , color=:blue)

plt3 = plot(dpi=300, title="Фазовый портрет", legend=false)
plot!(plt3,  $X_2$ ,  $Y_2$ , color=:red)

plt4 = plot(dpi=300, title="Решение уравнения", legend=false)
plot!(plt4,  $T$ ,  $X_2$ , color=:green)
plot!(plt4,  $T$ ,  $Y_2$ , color=:blue)

plt5 = plot(dpi=300, title="Фазовый портрет", legend=false)
plot!(plt5,  $X_3$ ,  $Y_3$ , color=:red)

plt6 = plot(dpi=300, title="Решение уравнения", legend=false)
plot!(plt6,  $T$ ,  $X_3$ , color=:green)
plot!(plt6,  $T$ ,  $Y_3$ , 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")

```

Рис. 5: Визуализация

Математические модели для первого случая(@fig:006 - @fig:007)

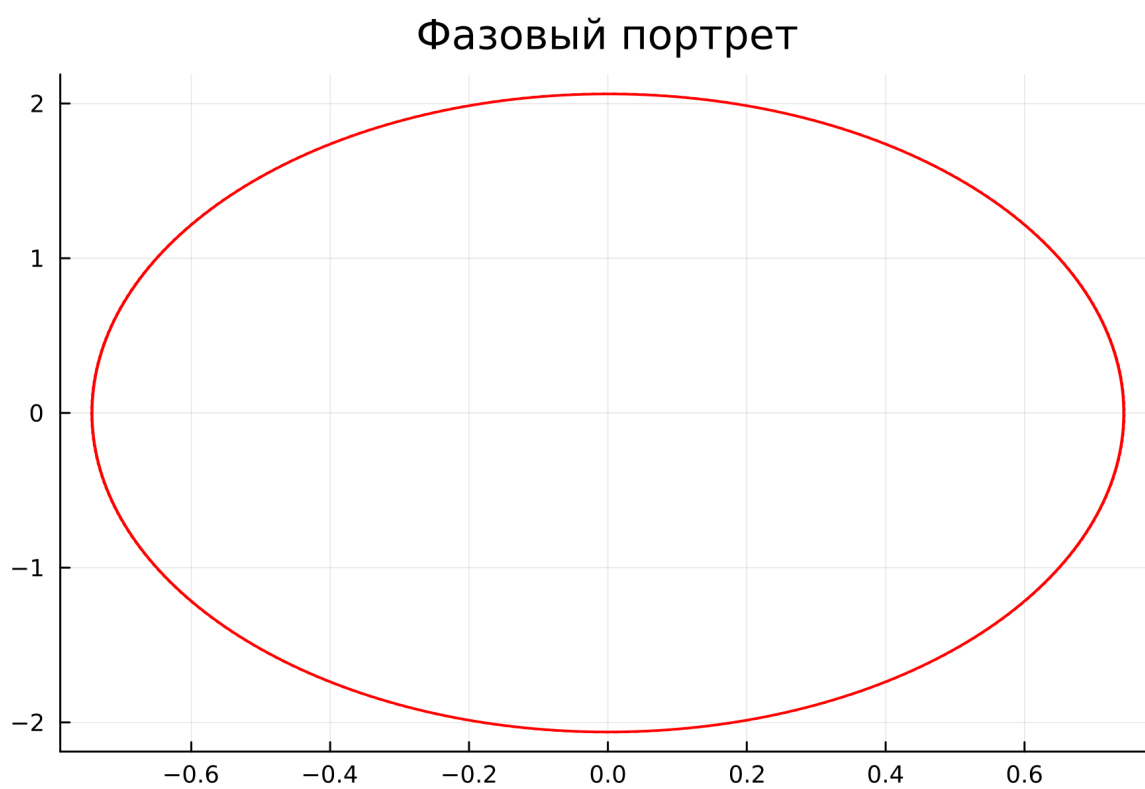


Рис. 6: Фазовый портрет первого случая

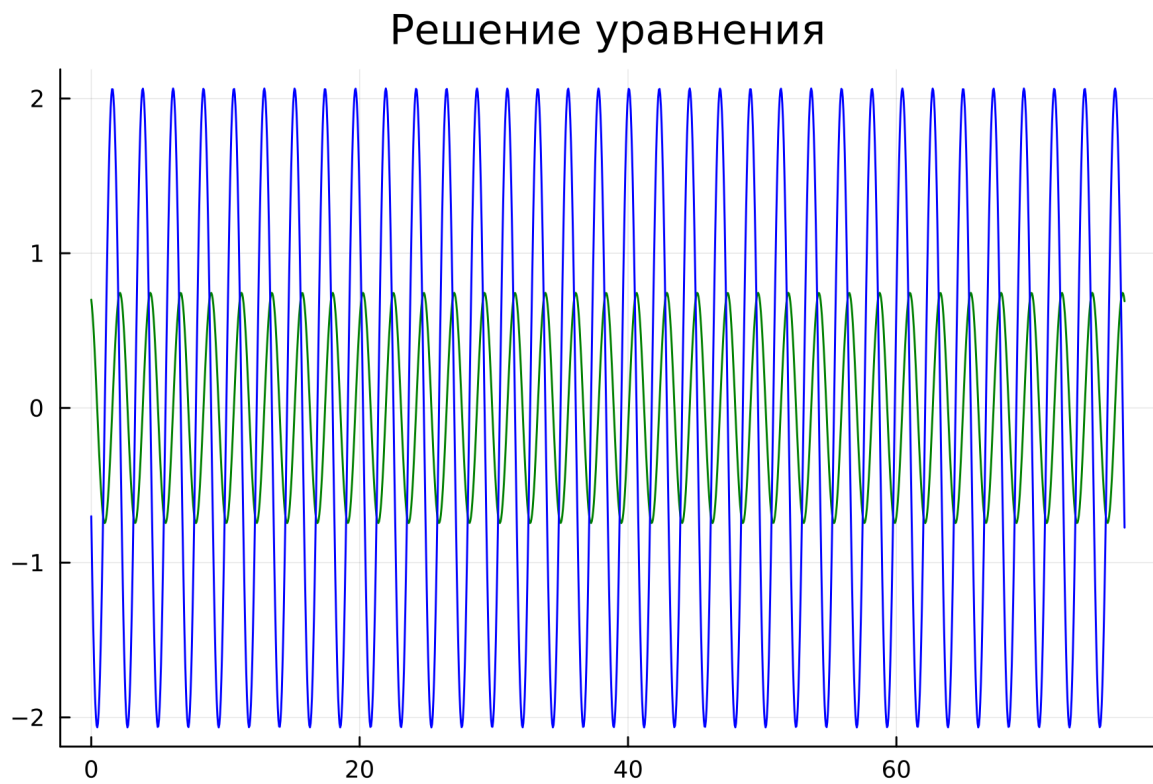


Рис. 7: Решение первого случая

Математические модели для второго случая(@fig:008 - @fig:009)

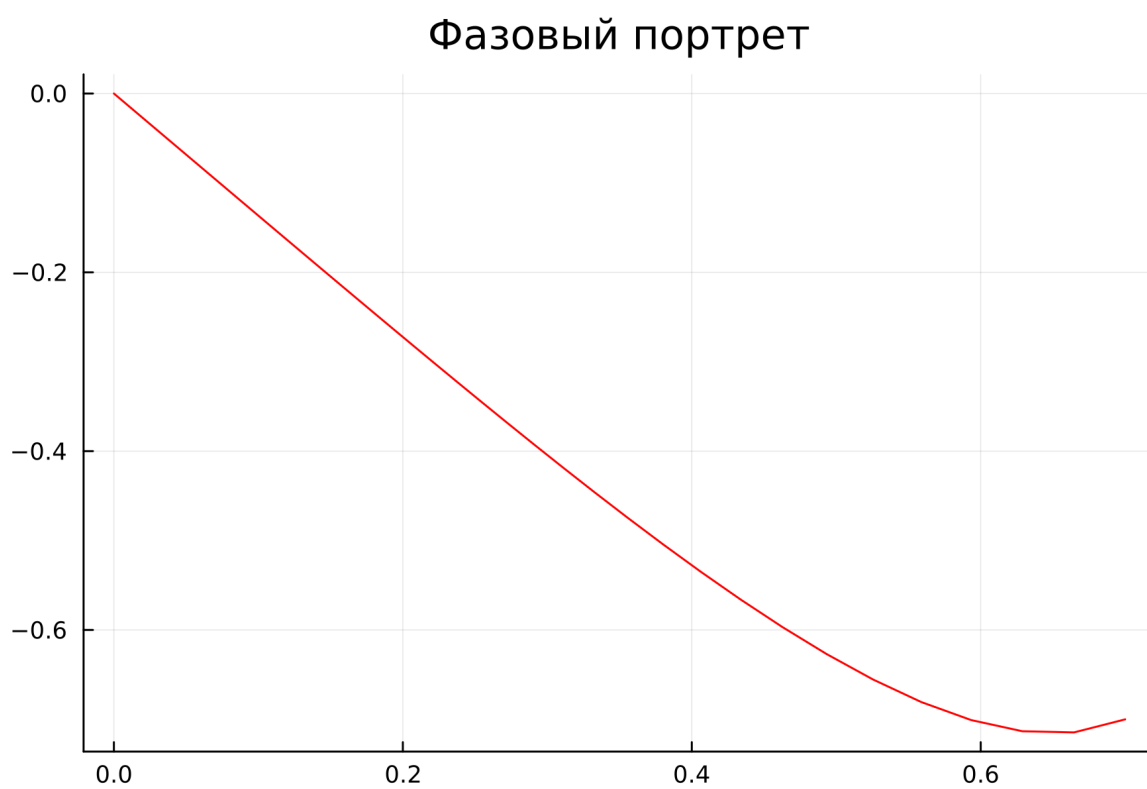


Рис. 8: Фазовый портрет второго случая

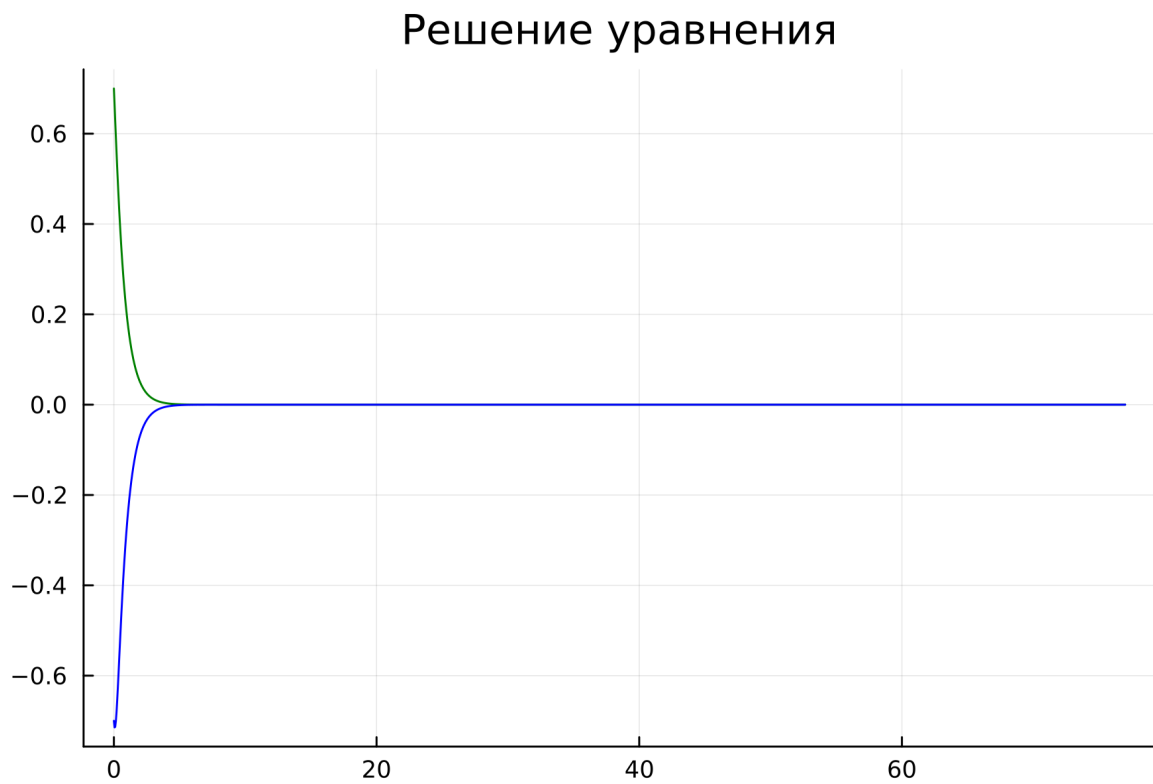


Рис. 9: Решение второго случая

Математические модели для третьего случая(@fig:008 - @fig:009)



Рис. 10: Фазовый портрет третьего случая

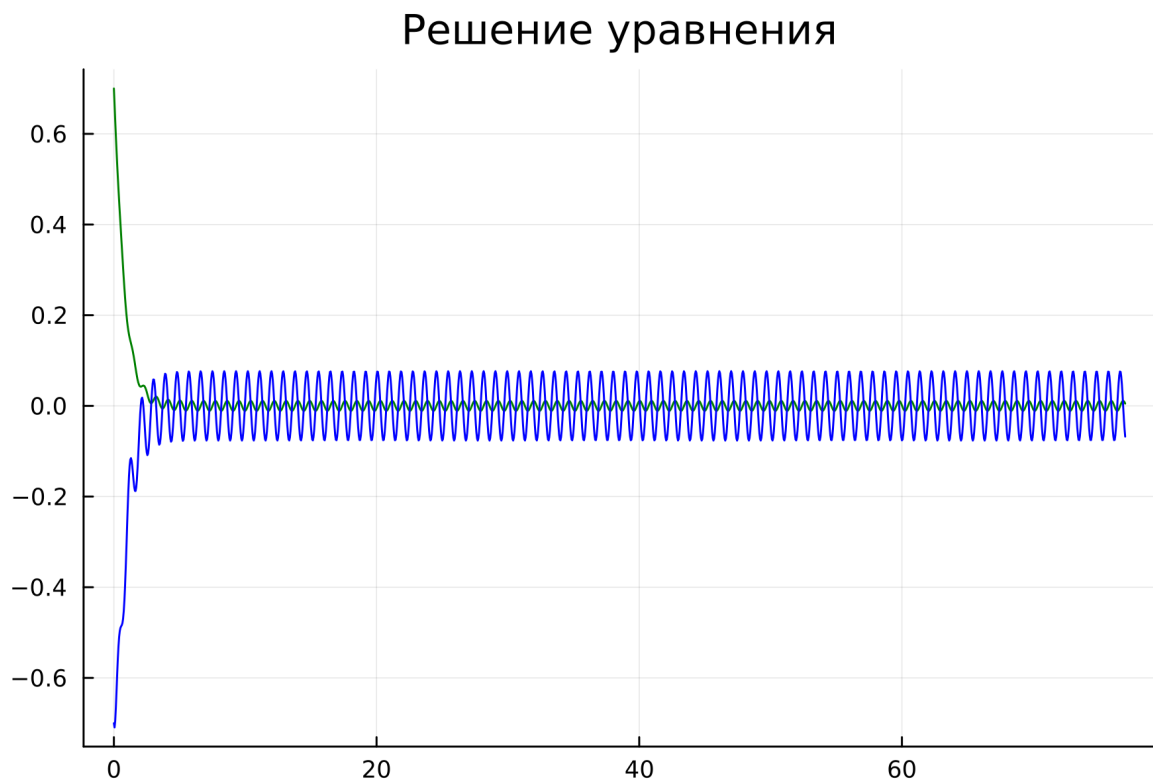


Рис. 11: Решение третьего случая

Код OpenModelica:

Сначала создадим X и Y , указав значений нулевых как стартовые, после чего опишем ω и γ . Далее запишем дифференциальное уравнение (@fig:012 - @fig:014).


```

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. 12: 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. 13: 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. 14: Код OpenModelica для третьего случая

Визуализируем фазовые портреты и решения на графике для первой случая (@fig:015 - @fig:016):

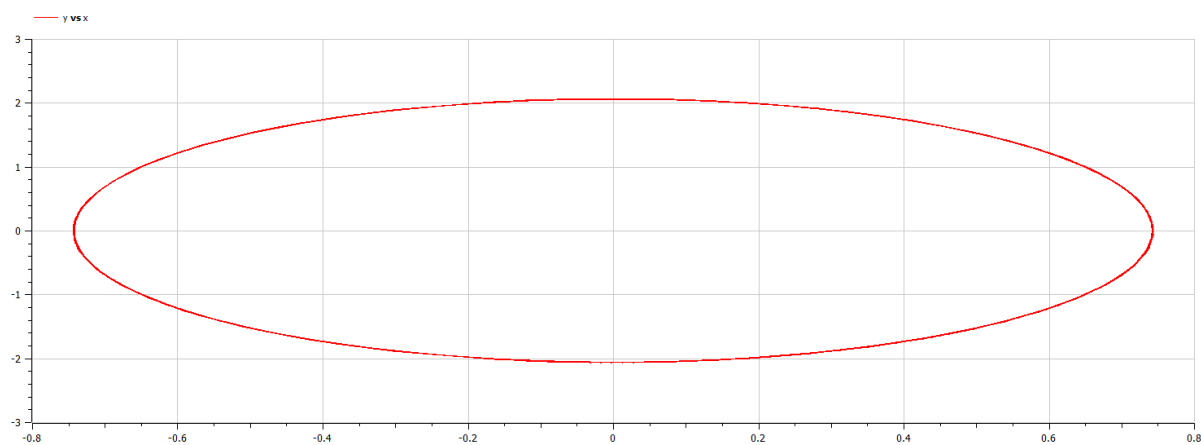


Рис. 15: Фазовый портрет первого случая

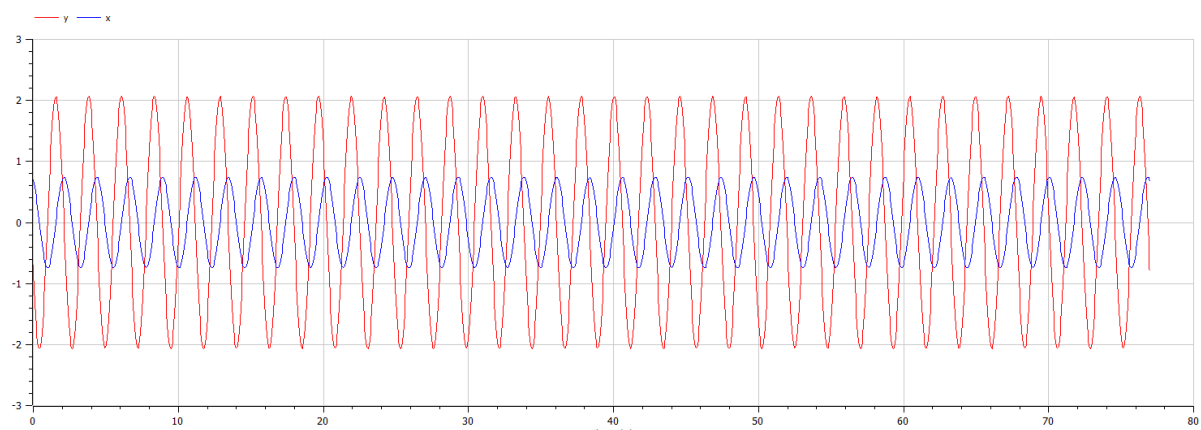


Рис. 16: Решение первого случая

Для второго случая (@fig:017 - @fig:018):

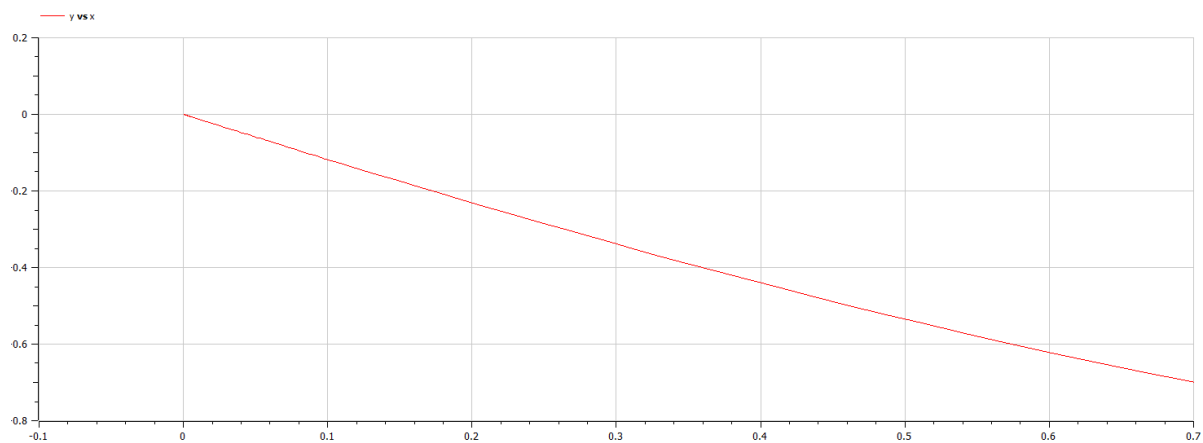


Рис. 17: Фазовый портрет второго случая

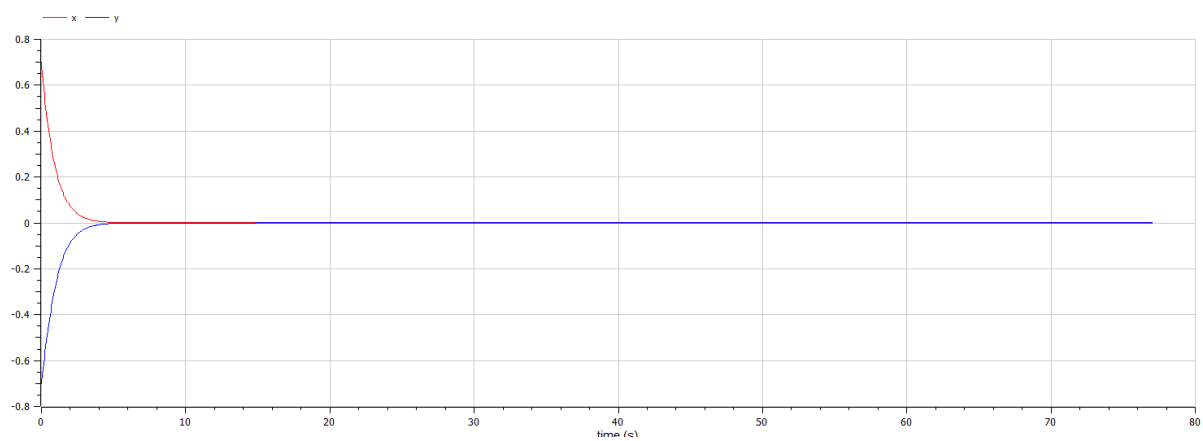


Рис. 18: Решение второго случая

Для третьего случая (@fig:019 - @fig:020):

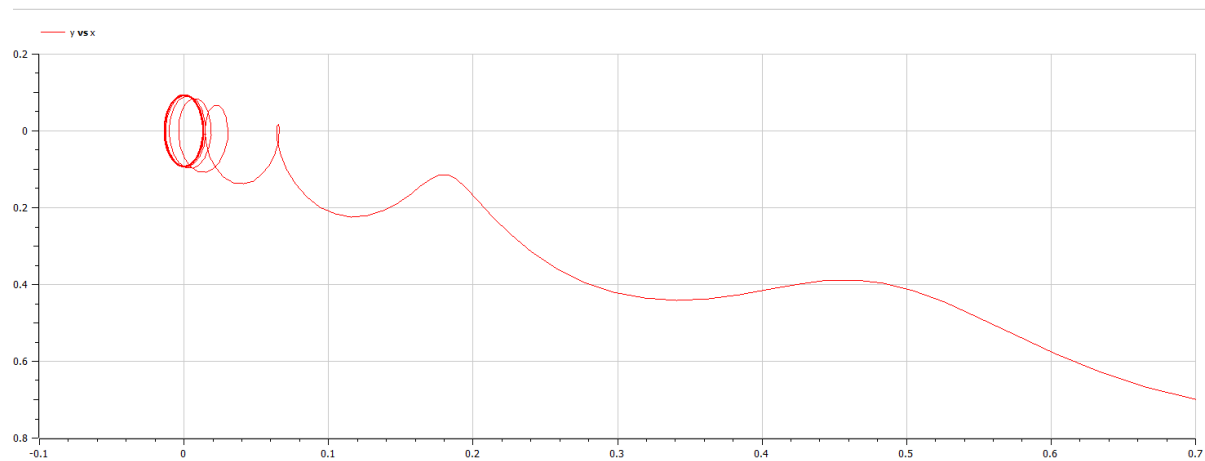


Рис. 19: Фазовый портрет третьего случая

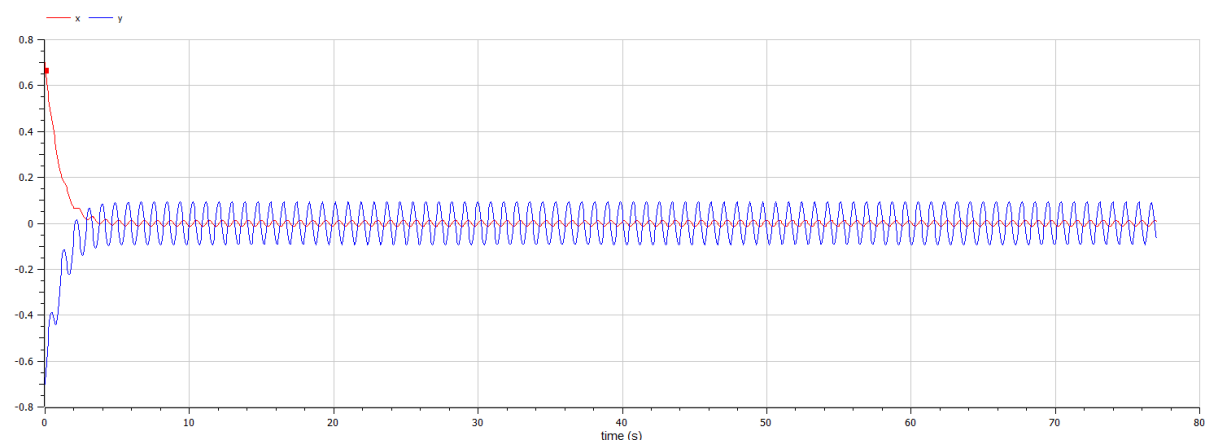


Рис. 20: Решение третьего случая

Вывод

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

Список Литературы

[1] <https://habr.com/ru/post/219337/>

[2] https://ru.wikipedia.org/wiki/Гармонический_осциллятор