

Лабораторная работа № 4

Тарусов Артём Сергеевич

2023, Москва

Целью данной работы является построение модели 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

Опишем начальные значения x и y , а также коэффициенты уравнения для первого случая согласно варианту 8 на языке Julia.

```
5      #Параметры осциллятора
6      #x'' + g * x' + w * x = f(t)
7      #w - квадрат частоты
8      #g - удвоенное затухание
9      w ::Float64 = 1.5
10     g ::Float64 = 0.0
11     x0 ::Float64 = 0.0
12     y0 ::Float64 = 0.0
```

Рис. 1: Начальные значения и коэффициенты для случая 1 на языке Julia

Опишем соответствующую систему дифференциальных уравнений.

```
15  function ode_fn(du, u, p, t)
16      x, y = u
17      du[1] = u[2]
18      du[2] = -w*u[1] - g*u[2]
19  end
```

Рис. 2: Система дифференциальных уравнений для случая 2 на языке Julia

Получим решение системы дифференциальных уравнений.

```
21     v0 ::Array{Any} = [x0, y0]
22     tspan = (0.0, 60.0)
23     prob = ODEProblem(ode_fn, v0, tspan)
24     sol = solve(prob, dtmax=0.05)
25
26     X = [u[1] for u in sol.u]
27     Y = [u[2] for u in sol.u]
28     T = [t for t in sol.t]
```

Рис. 3: Решение системы дифференциальных уравнений для случая 1 на языке Julia

Построим решение по полученным данным.

```
29 plt = plot(  
30     dpi=300,  
31     title="Решение уравнения",  
32     legend=false)  
33  
34 plot!(  
35     plt,  
36     T,  
37     X,  
38     color=:blue)  
39  
40 savefig(plt, "out/lab04_1_solve.png")
```

Рис. 4: Построение решения для случая 1 на языке Julia

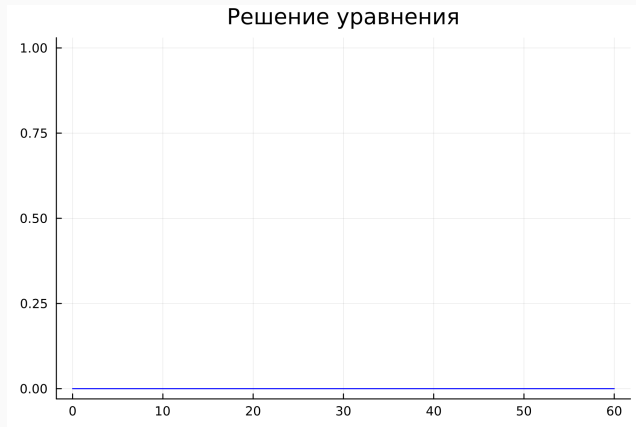


Рис. 5: Решение уравнения колебания гармонического осциллятора без затуханий и без действий внешней силы, построенное на Julia

Построим решение по полученным данным.

```
42 plt2 = plot(  
43     dpi=300,  
44     title="Фазовый портрет",  
45     legend=false)  
46  
47 plot!(  
48     plt2,  
49     X,  
50     Y,  
51     color=:blue)  
52  
53 savefig(plt2, "out/lab04_1_phase .png")
```

Рис. 6: Построение фазового портрета для случая 1 на языке Julia

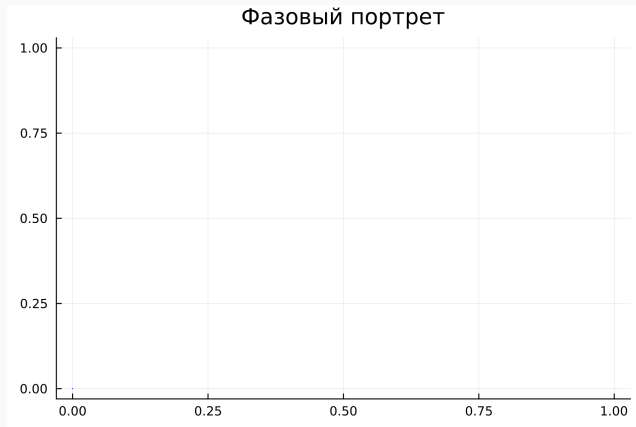


Рис. 7: Фазовый портрет колебания гармонического осциллятора без затуханий и без действий внешней силы, построенный на Julia

Для второго случая изменим значение коэффициентов.

```
5      #Параметры осциллятора
6      #x'' + g * x' + w * x = f(t)
7      #w - квадрат частоты
8      #g - удвоенное затухание
9      w ::Float64 = 10.0
10     g ::Float64 = 1.0
11     x0 ::Float64 = 0.0
12     y0 ::Float64 = 0.0
```

Рис. 8: Начальные значения и коэффициенты для случая 2 на языке Julia

Код решения остается без изменений. Получим решение и фазовый портрет.

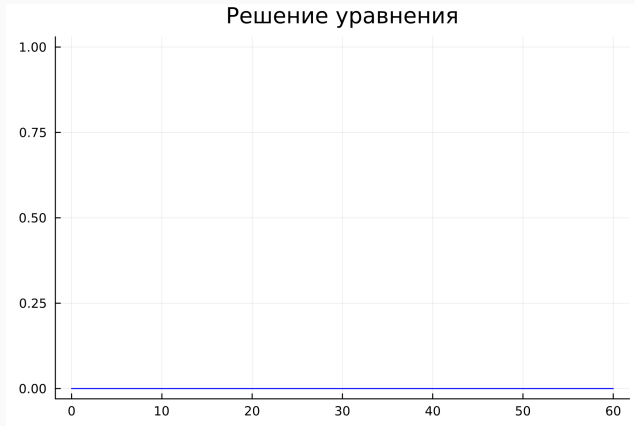


Рис. 9: Решение уравнения колебания гармонического осциллятора с затуханием и без действий внешней силы, построенное на Julia

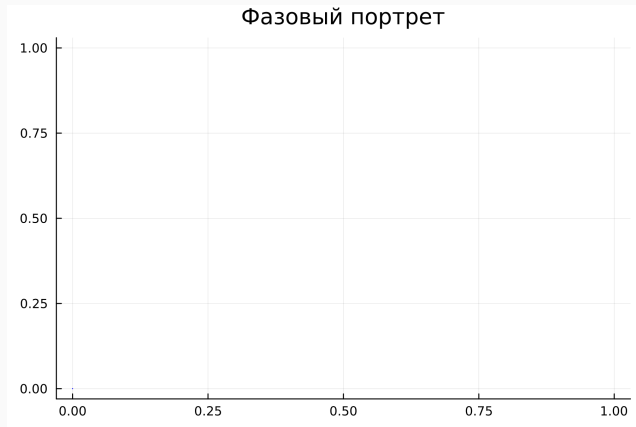


Рис. 10: Фазовый портрет колебания гармонического осциллятора с затуханием и без действий внешней силы, построенный на Julia

Изменим значение коэффициентов для третьего случая.

```
5      #Параметры осциллятора
6      #x'' + g * x' + w * x = f(t)
7      #w - квадрат частоты
8      #g - удвоенное затухание
9      w ::Float64 = 11.0
10     g ::Float64 = 1.0
11     x0 ::Float64 = 0.0
12     y0 ::Float64 = 0.0
```

Рис. 11: Начальные значения и коэффициенты для случая 3 на языке Julia

Добавим функцию внешней силы в систему дифференциальных уравнений.

```
15  function ode_fn(du, u, p, t)
16      x, y = u
17      du[1] = u[2]
18      du[2] = -w*u[1] - g*u[2] + 2*cos(t)
19  end
```

Рис. 12: Система дифференциальных уравнений для случая 3 на языке Julia

Получим решение и фазовый портрет.

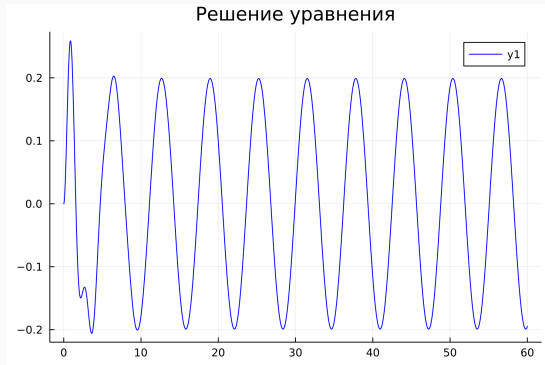


Рис. 13: Решение уравнения колебания гармонического осциллятора с затуханием и под действием внешней силы, построенное на Julia

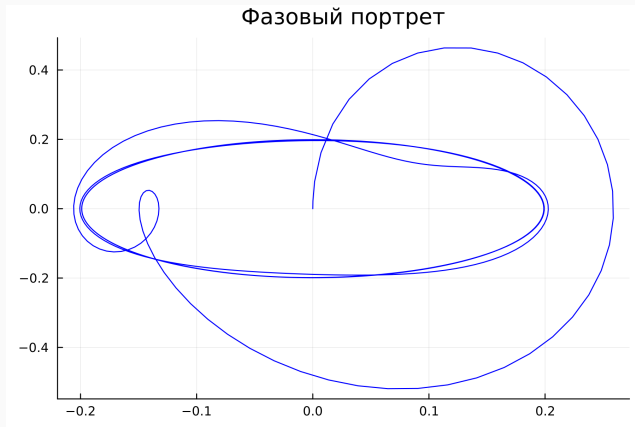


Рис. 14: Фазовый портрет колебания гармонического осциллятора с затуханием и под действием внешней силы, построенный на Julia

Построим модель для случая 1 на языке OpenModelica и получим решение и фазовый портрет.

```
1 model lab4_1
2 Real x;
3 Real y;
4 Real w = 1.5;
5 Real g = 0.0;
6 Real t = time;
7 initial equation
8 x = 0;
9 y = 0;
10 equation
11 der(x) = y;
12 der(y) = -w*x - g*y;
13 end lab4_1;
```

Рис. 15: Модель для случая 1 на языке OpenModelica

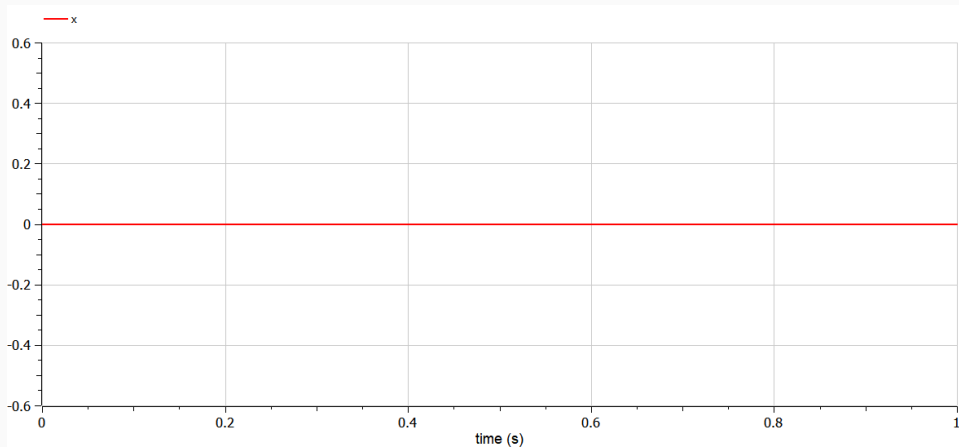


Рис. 16: Решение уравнения колебания гармонического осциллятора без затуханий и без действий внешней силы, построенное на OpenModelica

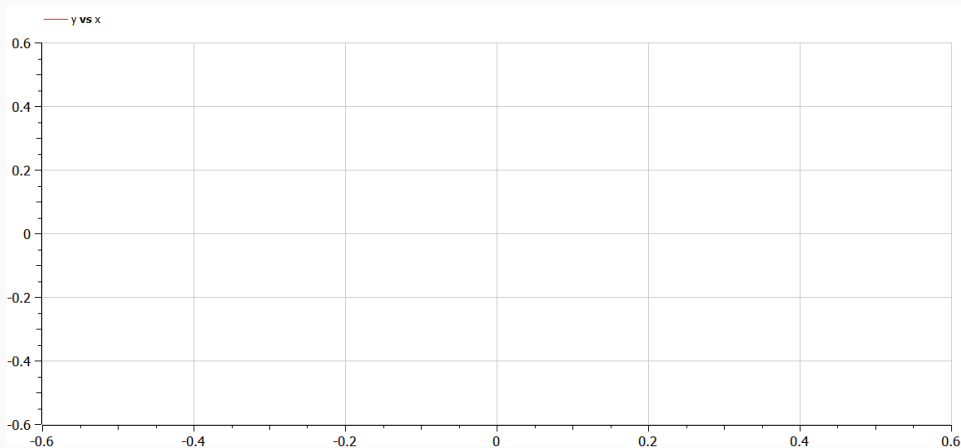


Рис. 17: Фазовый портрет колебания гармонического осциллятора без затуханий и без действий внешней силы, построенный на OpenModelica

Построим модель для случая 2 на языке OpenModelica и получим решение и фазовый портрет.

```
1  model lab4_2
2  Real x;
3  Real y;
4  Real w = 10.0;
5  Real g = 1.0;
6  Real t = time;
7  initial equation
8  x = 0;
9  y = 0;
10 equation
11 der(x) = y;
12 der(y) = -w*x - g*y;
13 end lab4_2;
```

Рис. 18: Модель для случая 2 на языке OpenModelica

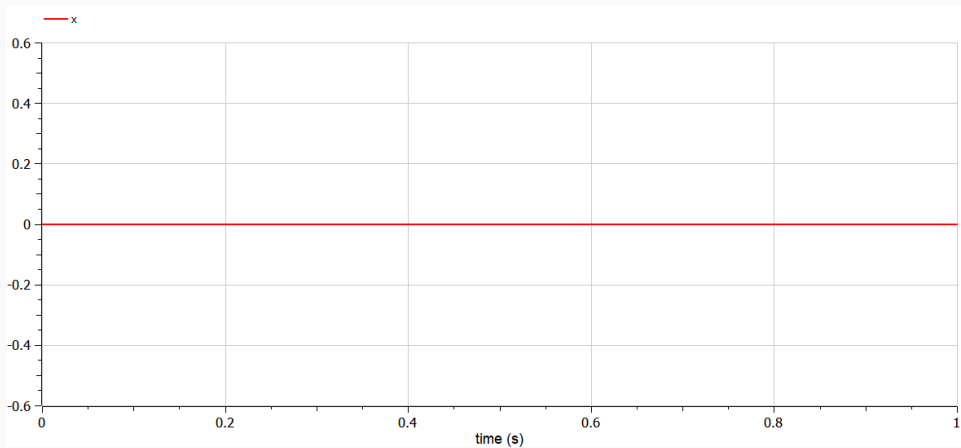


Рис. 19: Решение уравнения колебания гармонического осциллятора с затуханием и без действий внешней силы, построенное на OpenModelica

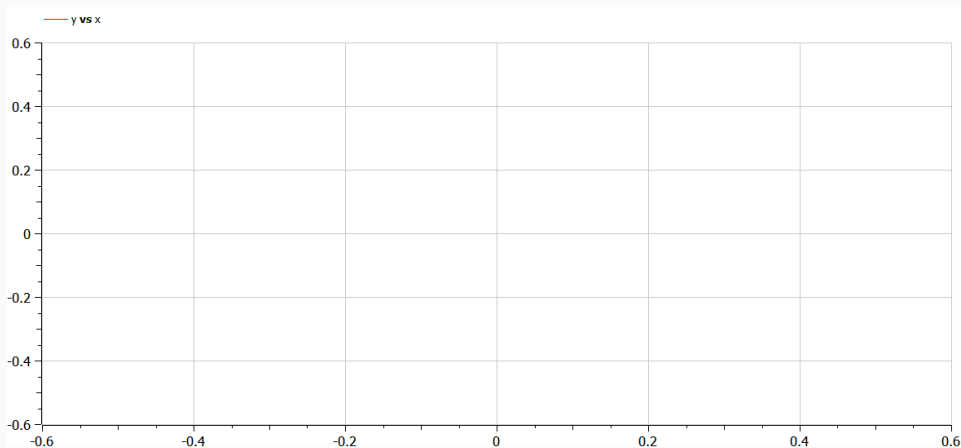


Рис. 20: Фазовый портрет колебания гармонического осциллятора с затуханием и без действий внешней силы, построенный на OpenModelica

Построим модель для случая 3 на языке OpenModelica и получим решение и фазовый портрет.

```
1  model lab4_3
2  Real x;
3  Real y;
4  Real w = 11.0;
5  Real g = 1.0;
6  Real t = time;
7  initial equation
8  x = 0;
9  y = 0;
10 equation
11 der(x) = y;
12 der(y) = -w*x - g*y + 2*cos(t);
13 end lab4_3;
```

Рис. 21: Модель для случая 3 на языке OpenModelica

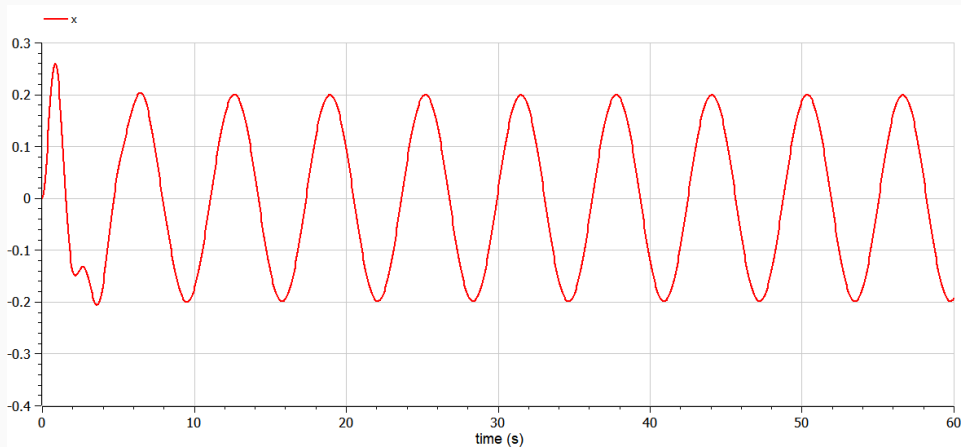


Рис. 22: Решение уравнения колебания гармонического осциллятора с затуханием и под действием внешней силы, построенное на OpenModelica

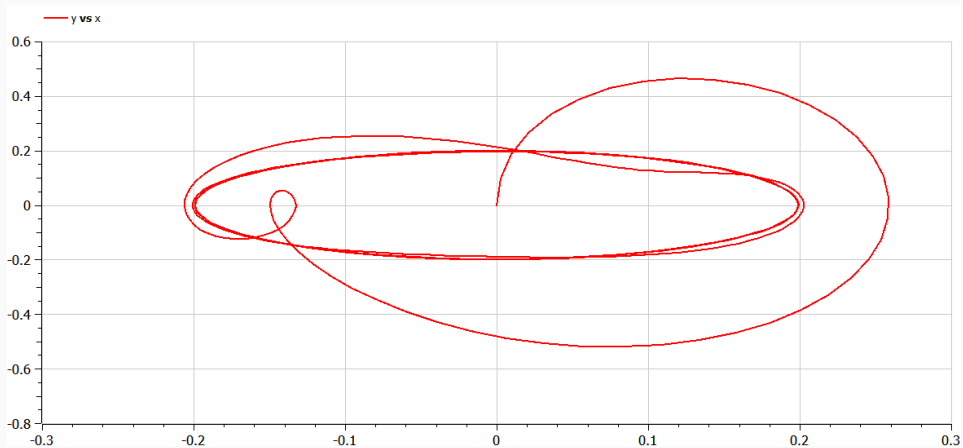


Рис. 23: Фазовый портрет колебания гармонического осциллятора с затуханием и под действием внешней силы, построенный на OpenModelica

В итоге проделанной работы мы построили по три модели на языках Julia и OpenModelica. Построение моделей колебания на языке openModelica занимает меньше строк, чем аналогичное построение на Julia.