

Control Theory: Assignment 1

Due on February 14, 2020 at 11:59pm

Mike Ivanov

Artem Bakhanov (B18-03)

1 Introduction

I created a GitHub repository [here](#). All the problems are solved by me, Artem Bakhanov, a student of Innopolis University. My variant is **j**.

2 Problem 2 (SO ODE)

$$4x'' - 4x' + 5t - 2x = 3, \quad x'(0) = 0, \quad x(0) = -3 \quad (1)$$

2.1 Simulink schema (without TF blocks)

Before transforming the equation to a Simulink schema I got the expression of x'' in terms of x' , x , and t :

$$x'' = x' + \frac{1}{2}x - \frac{5}{4}t + \frac{3}{4} \quad (2)$$

Initial conditions are 0 and -3 (from left to right).

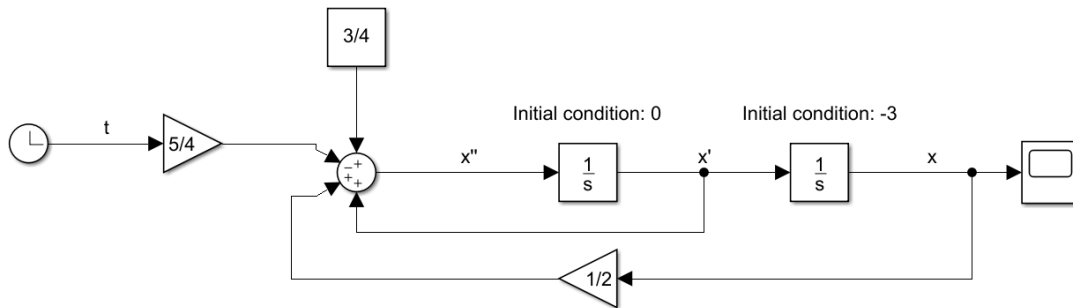


Figure 1: Simulink schema of the equation

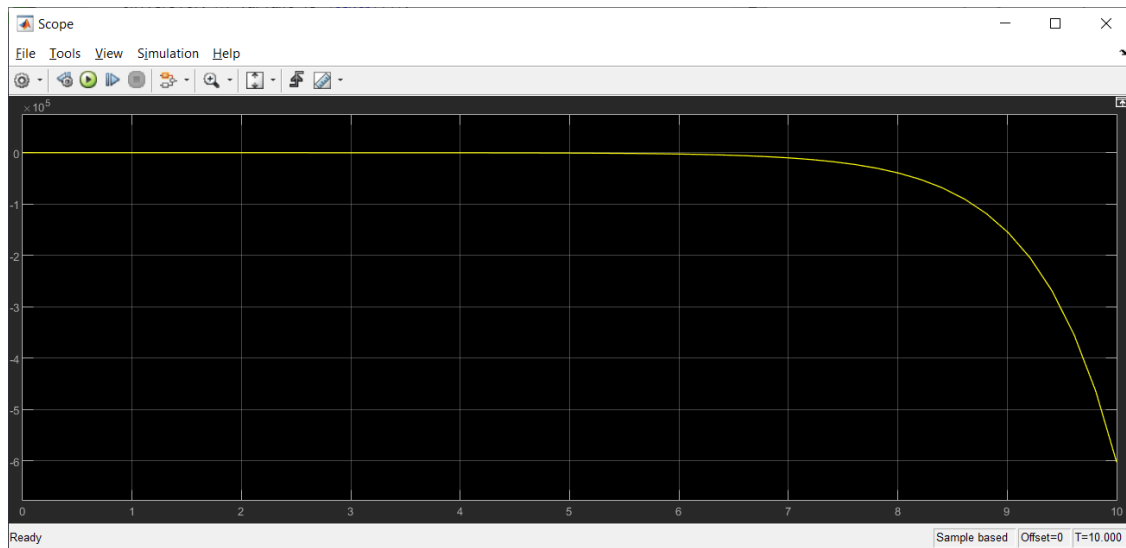


Figure 2: The plot of the solution

2.2 Simulink (with TF blocks)

2.3 Matlab solution of the ODE

You can also find this code in [task2c.j.m](#) file.

Listing 1: Solution of the ODE in Matlab

```
1
2 % define the function
3 syms x(t)
4
5 % create symbolic functions for initial conditions
6 Dx = diff(x, t);
7
8 % define the ode
9 ode = 4 * diff(x, t, 2) - 4 * diff(x, t) + 5 * t - 2 * x == 3;
10
11 % define initial conditions
12 cond1 = x(0) == -3;
13 cond2 = Dx(0) == 0;
14 conds = [cond1 cond2];
15
16 xSol(t) = dsolve(ode, conds);
17 disp(xSol);
18
19 % draw a plot
20 figure
21 fplot(xSol, [0 10]);
```

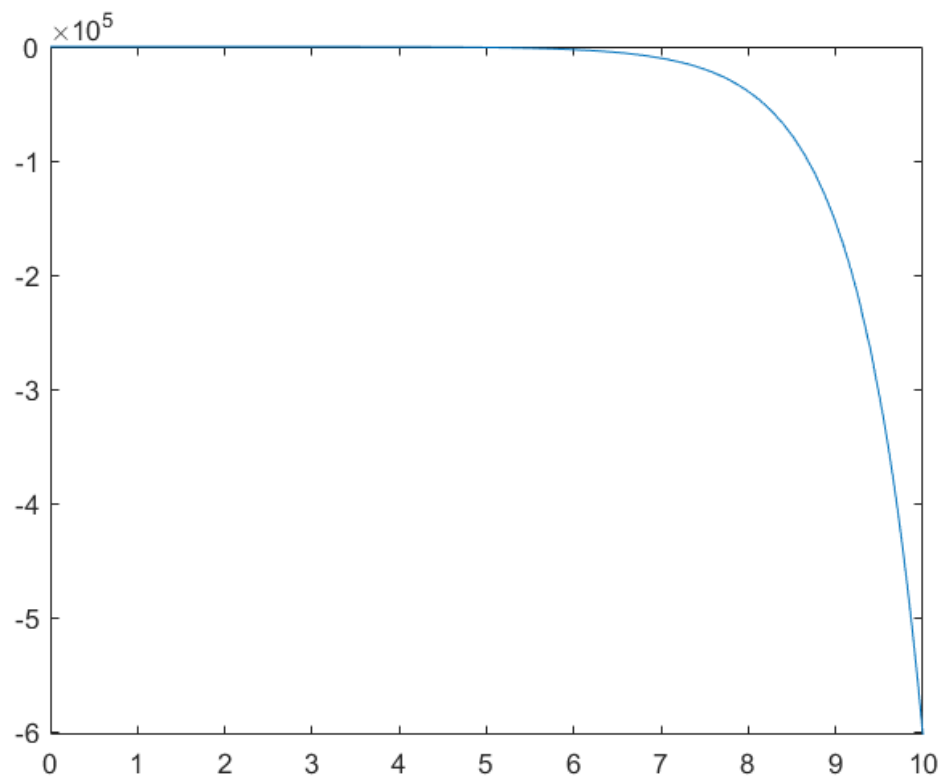


Figure 3: The plot of the Matlab solution

2.4 Laplace transform in Matlab

You can also find this code in [task2d_j.m](#) file.

Listing 2: Solution of the ODE using Laplace transform in Matlab

```

1 % define symbolic variables, X is Laplace transform of the solution
2 syms s t X;
3
4 % define x' and x''
5 X1 = s * X - (-3);
6 X2 = s * X1 - 0;
7
8 % define right-hand function and find its Laplace transform
9 f = 3 - 5 * t;
10 F = laplace(f);
11
12 % solve for X
13 Sol = solve(4*X2 - 4 * X1 - 2 * X - F, X);
14
15 % find the inverse Laplace transform of X
16 sol = ilaplace(Sol, s, t);
17
18 disp(sol);
19 % plot the solution graph
20 figure
21 fplot(sol, [0 10]);

```

3 ODE2SS (1)

Find the SS model of a system described by the following ODE:

$$x'' + 2x' - 3 = t + 5, \quad y = x' \quad (3)$$

The SS model will look like

$$\dot{x} = Ax + Bu \quad (4)$$

$$y = Cx + Du \quad (5)$$

Let $x = \begin{bmatrix} x \\ x' \end{bmatrix}$, $u = \begin{bmatrix} 1 \\ t \end{bmatrix}$. Let us express x'' :

$$x'' = 0x - 2x' + t + 8 \quad (6)$$

Then $A = \begin{bmatrix} 0 & 1 \\ 0 & -2 \end{bmatrix}$ and $B = \begin{bmatrix} 0 & 0 \\ 8 & 1 \end{bmatrix}$. It is easy to find matrices C and D. We are only interested in x' ; thus, $C = [0 \ 1]$ and $D = [0 \ 0]$. The final answer is:

$$\dot{x} = \begin{bmatrix} 0 & 1 \\ 0 & -2 \end{bmatrix} x + \begin{bmatrix} 0 & 0 \\ 8 & 1 \end{bmatrix} u \quad (7)$$

$$y = [0 \ 1] x + [0 \ 0] u \quad (8)$$

4 ODE2SS (2)

Find the SS model of a system described by the following ODE:

$$x'''' + 3x''' + 4x'' + 2x' - 6 = 2u_1 + 2u_2, \quad y = x' + u_1 + u_2 \quad (9)$$

The SS model will look like

$$\dot{x} = Ax + Bu \quad (10)$$

$$y = Cx + Du \quad (11)$$

Let $x = \begin{bmatrix} x \\ x' \\ x'' \\ x''' \end{bmatrix}$, $u = \begin{bmatrix} 1 \\ u_1 \\ u_2 \end{bmatrix}$. Then, obviously, $\dot{x} = \begin{bmatrix} x' \\ x'' \\ x''' \\ x'''' \end{bmatrix}$. Let us express x'''' :

$$x'''' = 0x - 2x' - 4x'' - 3x''' + 6 + 2u_1 + 2u_2 \quad (12)$$

Then

$$A = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & -2 & -4 & -3 \end{bmatrix}, \quad B = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \\ 6 & 2 & 2 \end{bmatrix} \quad (13)$$

and

$$C = [0 \quad 1 \quad 0 \quad 0], \quad D = [0 \quad 1 \quad 1] \quad (14)$$

The final answer is:

$$\dot{x} = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & -2 & -4 & -3 \end{bmatrix} x + \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \\ 6 & 2 & 2 \end{bmatrix} u \quad (15)$$

$$y = [0 \quad 1 \quad 0 \quad 0] x + [0 \quad 1 \quad 1] u \quad (16)$$

5 ODE2SS (Python)

The code is available in the [problem5.ipynb](#) file. You need to have Jupyter Notebook installed on your computer to run the notebook, NumPy library is also required.

6 ODEs and SSs (Python)