Digital Control

Dimitrios Nentidis

November 2023

1 Dynamical System Description

The system comprises seven state variables, $x_1, x_2, x_3, x_4, x_5, x_6, x_7$, and is influenced by three control inputs, u_1, u_2, u_3. It exhibits non-linear dynamics with intricate interactions among the state variables. The output signals, y_1, y_2, y_3, are formed as linear combinations of the state variables. This system's complexity arises from its non-linearities and the inter-dependencies among its components.

$$\dot{x}_1(s) = x_2 + 5x_1^2 x_2 + 6x_2^2 \tag{1}$$
$$\dot{x}_2(s) = -4x_1 - 5x_2 + 8x_1 x_2 + u_1 \tag{2}$$
$$\dot{x}_3(s) = x_4 \tag{3}$$
$$\dot{x}_4(s) = -6x_3 - 5x_4 + 3x_3^3 + 10x_3 x_4 x_5 + u_2 \tag{4}$$
$$\dot{x}_5(s) = x_6 + 4x_7^2 \tag{5}$$
$$\dot{x}_6(s) = x_7 + 5x_5 x_6^2 x_7 \tag{6}$$
$$\dot{x}_7(s) = -14x_5 - 23x_6 - 10x_7 + 7x_5 x_6 x_7 + u_3 \tag{7}$$
$$y_1 = 5x_1 + 5x_2 + 6x_3 + 2x_4 + 14x_5 + 9x_6 + x_7 \tag{8}$$
$$y_2 = 8x_1 + 2x_2 + 3x_2 + 4x_5 + 6x_6 + 2x_7 \tag{9}$$
$$y_3 = x_1 + x_2 + 4x_3 + 2x_4 + 1.4x_5 + 0.2x_6 \tag{10}$$

2 Linearization

The first step to tackle the control of the above system is to linearize it. For this, Matlab's symbolic toolbox was used. The system is defined as follows:

Listing 1: System is defined in Matlab.

```
% Define symbolic variables
syms x1 x2 x3 x4 x5 x6 x7 u1 u2 u3

% Define the system equations
x1_dot = x2 + 5*x1^2 * x2 + 6*x2^2;
x2_dot = -4*x1 - 5*x2 + 8*x1*x2 + u1;
x3_dot = x4;
x4_dot = -6*x3 - 5*x4 + 3*x3^3 + 10*x3*x4*x5 + u2;
x5_dot = x6 + 4*x7^2;
```

```
x6_dot = x7 + 5*x5 * x6^2 * x7;
x7_dot = -14*x5 - 23*x6 - 10*x7 + 7*x5 * x6 * x7 + u3;
```

Then using the equilibrium point the system is linearized and the following linearized matrices are obtained.

Linearized System Matrix A:

$$A = \begin{bmatrix} 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ -4 & -5 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & -6 & -5 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & -14 & -23 & -10 \end{bmatrix}$$

Linearized Input Matrix B:

$$B = \begin{bmatrix} 0 & 0 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

Output Matrix C:

$$C = \begin{bmatrix} 5 & 5 & 6 & 2 & 14 & 9 & 1 \\ 8 & 2 & 3 & 0 & 4 & 6 & 2 \\ 1 & 1 & 4 & 2 & \frac{7}{5} & \frac{1}{5} & 0 \end{bmatrix}$$

Feedforward Matrix D:

$$D = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix}$$

3 Transfer Functions

Using the ss2tf matlab function the following transfer functions are obtained for output y_1:

$$\frac{y_1}{u_1} = \exp(-2s)*\frac{5s^6 + 80s^5 + 470s^4 + 1340s^3 + 1985s^2 + 1460s + 420}{s^7 + 20s^6 + 158s^5 + 644s^4 + 1469s^3 + 1880s^2 + 1252s + 336}$$

$$\frac{y_2}{u_1} = \exp(-2s)*\frac{2s^6 + 38s^5 + 278s^4 + 1010s^3 + 1928s^2 + 1832s + 672}{s^7 + 20s^6 + 158s^5 + 644s^4 + 1469s^3 + 1880s^2 + 1252s + 336}$$

$$\frac{y_3}{u_1} = \exp(-2s)*\frac{s^6 + 16s^5 + 94s^4 + 268s^3 + 397s^2 + 292s + 84}{s^7 + 20s^6 + 158s^5 + 644s^4 + 1469s^3 + 1880s^2 + 1252s + 336}$$

For y_2:

$$\frac{y_1}{u_2} = \exp(-2s)*\frac{2s^6 + 36s^5 + 244s^4 + 800s^3 + 1338s^2 + 1084s + 336}{s^7 + 20s^6 + 158s^5 + 644s^4 + 1469s^3 + 1880s^2 + 1252s + 336}$$

$$\frac{y_2}{u_2} = \exp(-2s)*\frac{3s^5 + 45s^4 + 231s^3 + 507s^2 + 486s + 168}{s^7 + 20s^6 + 158s^5 + 644s^4 + 1469s^3 + 1880s^2 + 1252s + 336}$$

$$\frac{y_3}{u_2} = \exp(-2s)*\frac{2s^6 + 34s^5 + 214s^4 + 646s^3 + 1000s^2 + 760s + 224}{s^7 + 20s^6 + 158s^5 + 644s^4 + 1469s^3 + 1880s^2 + 1252s + 336}$$

For y_3:

$$\frac{y_1}{u_3} = \exp(-1s)*\frac{s^6 + 19s^5 + 139s^4 + 505s^3 + 964s^2 + 916s + 336}{s^7 + 20s^6 + 158s^5 + 644s^4 + 1469s^3 + 1880s^2 + 1252s + 336}$$

$$\frac{y_2}{u_3} = \exp(-1s)*\frac{2s^6 + 26s^5 + 134s^4 + 350s^3 + 488s^2 + 344s + 96}{s^7 + 20s^6 + 158s^5 + 644s^4 + 1469s^3 + 1880s^2 + 1252s + 336}$$

$$\frac{y_3}{u_3} = \exp(-1s)*\frac{0.2s^5 + 3.4s^4 + 21s^3 + 59s^2 + 74.8s + 33.6}{s^7 + 20s^6 + 158s^5 + 644s^4 + 1469s^3 + 1880s^2 + 1252s + 336}$$

The step response of these can be found below.

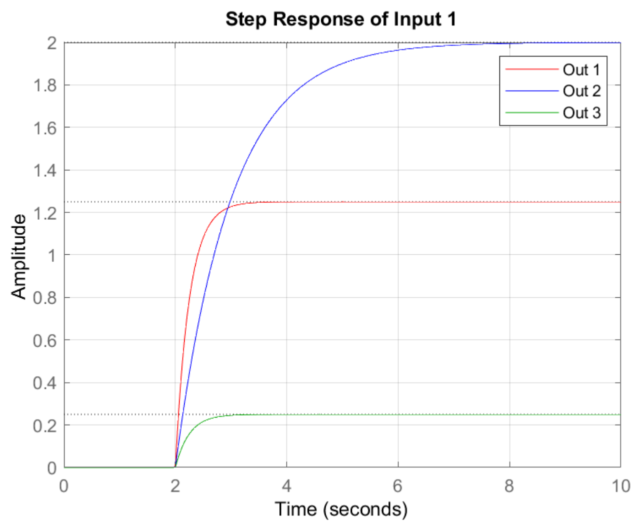

Figure 1: Step response o u_1.

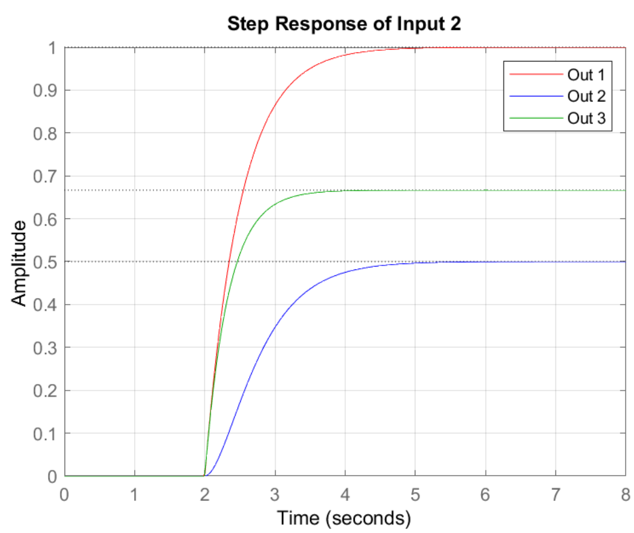

Figure 2: Step response of u_2.

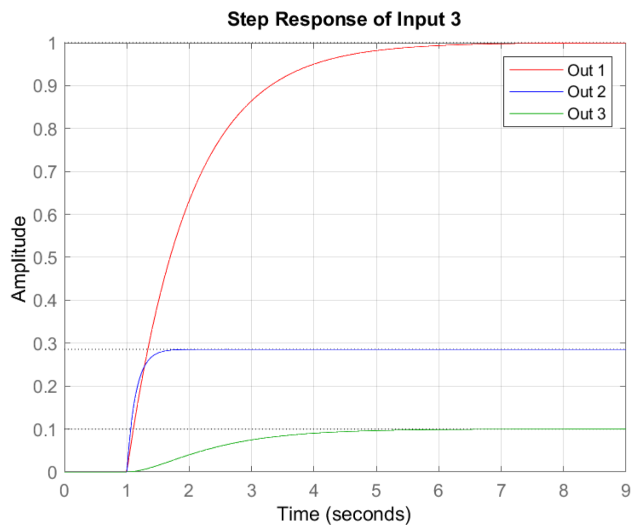

Figure 3: Step response of u_3.

4 Relative Gain Array and Controlability

To compute the Relative Gain Array (RGA), the gain matrix K needs to be computed first. This involves the calculation of the steady-state gain for each input-output transfer function, a fundamental step in the analysis.

$$K = \begin{bmatrix} 1.25 & 1 & 1 \\ 2 & 0.5 & 0.286 \\ 0.25 & 0.667 & 0.1 \end{bmatrix}$$

$$RGA = K \cdot (K^{-1})'$$

The following is the Relative Gain Array.

$$RGA = \begin{bmatrix} -0.1945 & -0.1421 & 1.3366 \\ 1.2537 & -0.0691 & -0.1846 \\ -0.0591 & 1.2112 & -0.1520 \end{bmatrix}$$

Using the RGA matrix, the inputs are paired with the outputs as follows.
$y_2 \rightarrow u_1$
$y_3 \rightarrow u_2$
$y_1 \rightarrow u_3$

To check for the controllability of the system the following matrix is obtained. If the determinant is non zero and the matrix is full rank, then the system is controllable.

$$C_o = \begin{bmatrix} A & AB & \dots & A^{n-1}B \end{bmatrix}$$

In our case the system is indeed controllable

5 Controlling the system

A model of the system was created in simulink to aid with the tuning of the PIDs used for control. The model can be seen below.

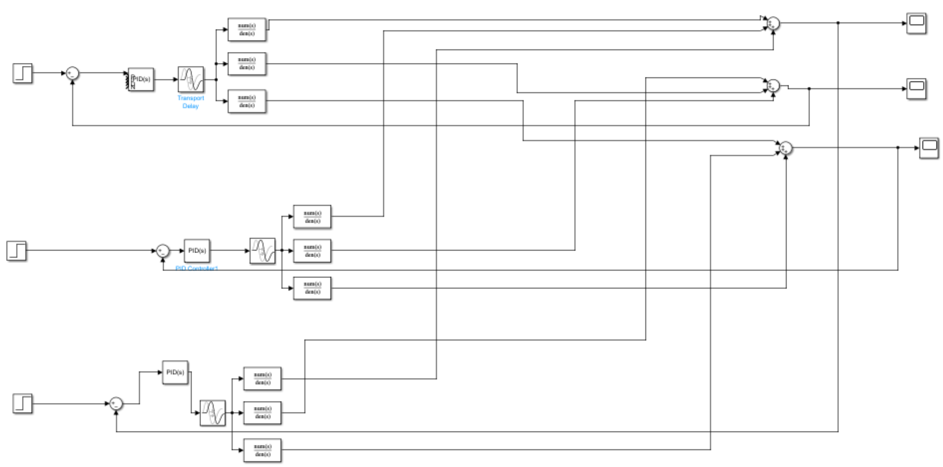

Figure 4: Simulink Model.

Using Simulink's Controller Design Toolbox, the three parallel form PIDs are tuned the following gains:

For $y_2 \rightarrow u_1$ PID controller:

$$\begin{aligned} \text{Proportional Gain (P)} &= 0.114 \\ \text{Integral Gain (I)} &= 0.157 \\ \text{Derivative Gain (D)} &= 0 \\ \text{Filter Coefficient (N)} &= 100 \end{aligned}$$

For $y_3 \rightarrow u_2$ PID controller:

$$\begin{aligned}\text{Proportional Gain (P)} &= 0.644\\ \text{Integral Gain (I)} &= 0.557\\ \text{Derivative Gain (D)} &= -0.497\\ \text{Filter Coefficient (N)} &= 0.394\end{aligned}$$

For $y_1 \rightarrow u_3$ PID controller:

$$\begin{aligned}\text{Proportional Gain (P)} &= 0.820\\ \text{Integral Gain (I)} &= 0.531\\ \text{Derivative Gain (D)} &= -0.009\\ \text{Filter Coefficient (N)} &= 87.443\end{aligned}$$

For a set point of +0.2 applied to y_1, a step input of the paired input u_3 was provided. The figure below illustrates the responses of all outputs.

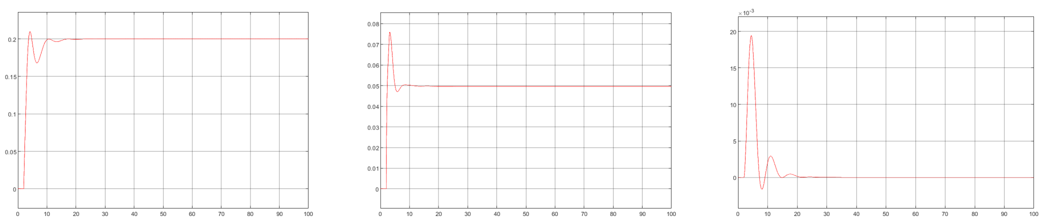

Figure 5: y_1, y_2, and y_3.

For set points of +0.2 applied to y_1 and -0.2 applied to y_2, corresponding step inputs are applied to u_3 and u_1, respectively.

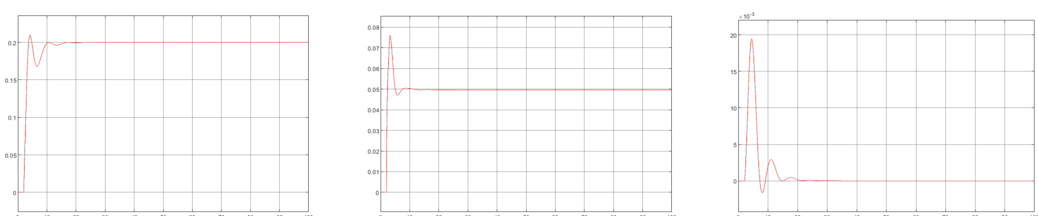

Figure 6: y_1, y_2, and y_3.

For set points of +0.2 applied to y_1, -0.2 applied to y_2 and +0.1 applied to y_3, corresponding step inputs are applied to u_3, u_1 and u_2 respectively.

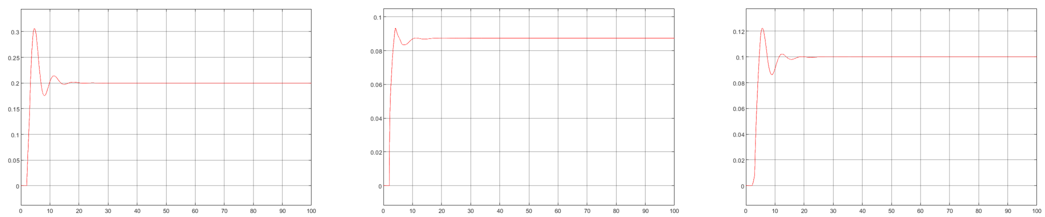

Figure 7: y_1, y_2, and y_3.

Lastly adding the filter to the model as seen below and inducing the same step input as before, +0.2 applied to y_1, -0.2 applied to y_2 and +0.1 applied to y_3, corresponding step inputs are applied to u_3, u_1 and u_2 respectively.

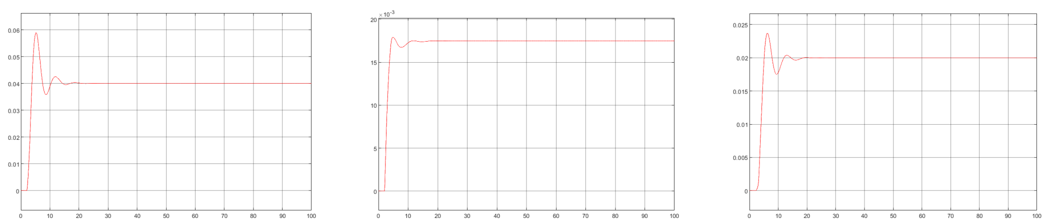

Figure 8: y_1, y_2, and y_3 with the filter.

Noteworthy is that the filter reduces the amplitude oscillations.

6 Decoupling

Decoupling a coupled MIMO system involves creating a decoupler matrix that transforms the coupled system into a set of independent SISO systems when pre-multiplied with the input vector. The decoupler attempts to cancel out the interactions between non-paired inputs and outputs.

The decoupler is designed based on the inverse of the steady-state gain matrix. If K is the steady-state gain matrix, the ideal decoupler D would be

$$D = K^{-1}$$

which when applied to the system, would ideally negate non-desired interactions. For this system, the D matrix turns out to be the following:

$$K = \begin{bmatrix} -0.1556 & 0.6268 & -0.2366 \\ -0.1421 & -0.1382 & 1.8158 \\ 1.3366 & -0.6454 & -1.5201 \end{bmatrix}$$

This was inserted into the simulink model as gain blocks in each corresponding input-output relationship. Decoupling the system without altering the PID parameters resulted to instabilities. To counter them the PIDs were retuned using the simulink toolbox but to no avail. Below both attempted implementations as weel as one of the controller's output can be found.

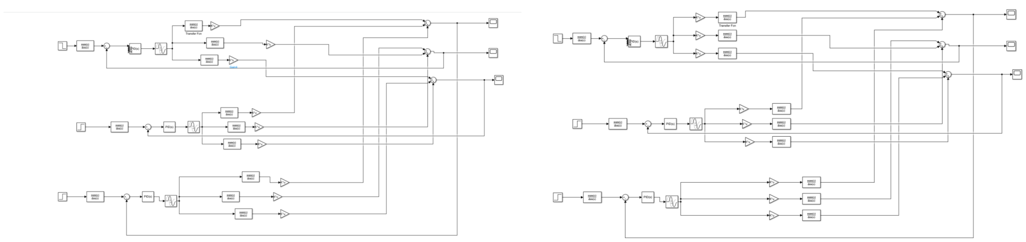

Figure 9: Two attempted implementations.

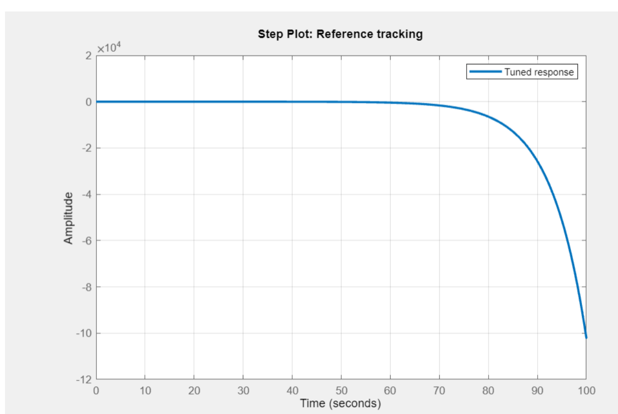

Figure 10: PID tuning.

Theoretically, if the model worked, to check the efficiency of the decoupled system, one can compare the step response of each output when their paired input is excited alone to when all the inputs are excited. If the responses are similar then the decoupler implementation is a good addition.