Ball & plate system control

Seyed Habib Hoseini
Mehdi Fayaz

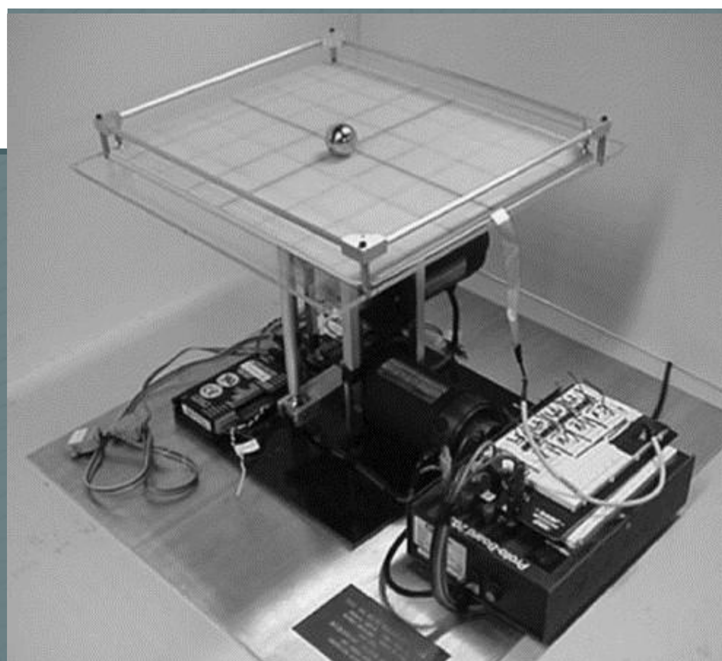
July 7, 2024

Modern Control

Prof. Hamid D.Taghirad

INTRODUCTION

The ball and plate system is a classic control engineering problem that is an excellent platform for applying modern control theories.
This system involves controlling the position of a ball on a flat plate by adjusting the plate's tilt. The primary objective is stabilizing the ball at a desired position using advanced control techniques based on state-space representation and state feedback.



THE PROCESS

1. System Components:

***Plate**:* A flat surface where the ball moves.
***Ball**:* The object whose position is to be controlled.
***Actuators**:* Motors or servos to tilt the plate.
***Sensors**:* Devices to detect the position of the ball, such as cameras or potentiometers.
***Controller**:* A microcontroller or a computer to process the sensor data and control the actuators.

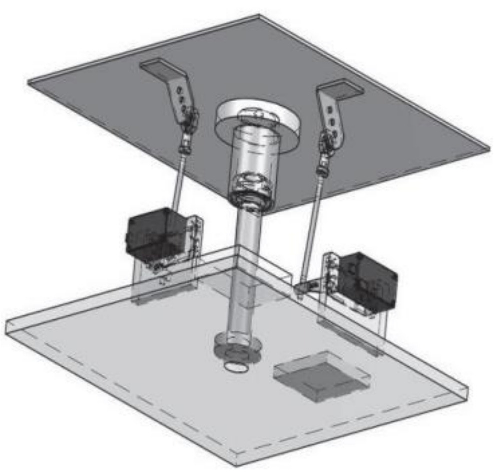


Figure 1. Mechanical design of the BaP system. system.

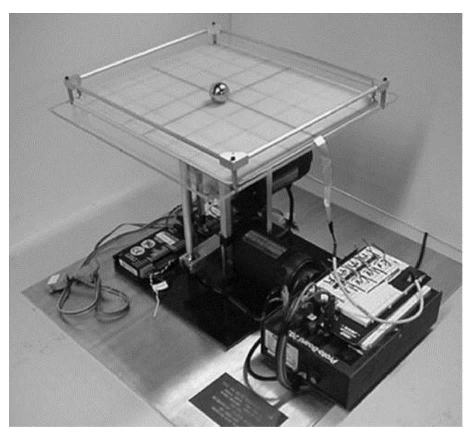


Figure 2. Final view of the BaP

2. Mathematical Modeling:

Develop the mathematical model of the system. This involves the dynamics of the ball and the plate. Use Newton's laws to derive the equations of motion for the ball on the plate.

3. Control Strategy:

Choose a control strategy like PID (Proportional-Integral-Derivative) control, LQR (Linear Quadratic Regulator), or a more advanced control method. Design the controller to process the ball’s position data and adjust the plate’s tilt to keep the ball at the desired position.

4. Simulation:

Simulate the system using software tools like MATLAB/Simulink. Verify the stability and performance of your control strategy through simulations before implementing it in the real world.

5. Implementation:

Implement the control algorithm on your hardware setup. Ensure real-time processing capability if using a microcontroller. Test the system in real scenarios and fine-tune the controller parameters as needed.

6. Testing and Calibration:

Test the system thoroughly to handle different disturbances and initial conditions. Calibrate the sensors and actuators for accurate measurements and movements.

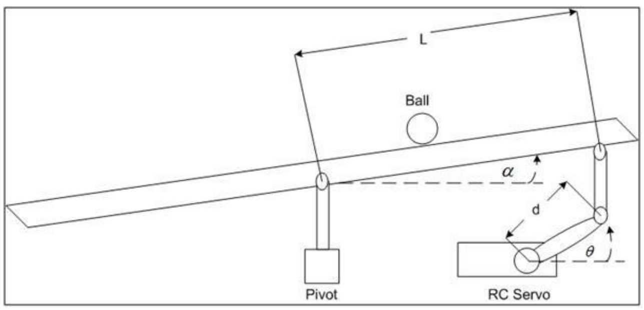
Mathematical modeling and obtaining the state space model

The ball and plate system is divided into two main parts: the mechanical part and the electrical part. This section provides the dynamic equations for both parts and combines them to derive the general state-space equations.

1. Mechanical Part

The mechanical part of the system involves the dynamics of the ball rolling on the plate. The key variables are the position of the ball (x,y) on the plate and the tilt angles of the plate (θ_x, θ_y).

First, we examine one direction and generalize the other direction:



x : position
$\dot{x}$: velocity
$\ddot{x}$: Acceleration
r : radius of the ball

Figure 3. Mechanical Part.

$$\left(\frac{J_b}{r^2} + m\right)\ddot{x}(t) + mg\,sin\,\alpha(t) - mx(t)\dot{\alpha}(t) + \frac{J_b}{r}\ddot{\alpha}(t) = 0 \qquad (1)$$

If we assume that α is small:

$$sin\,\alpha \approx \alpha\,, \dot{\alpha} = 0\,, \ddot{\alpha} = 0 \qquad (2)$$

Considering (2) and knowing that $J_b = \frac{2}{3}mr^2$ we rewrite the equation as below:

$$-\left(\frac{2mr^2}{3r^2} + m\right)\ddot{x}(t) + mg\,sin\,\alpha(t) = 0 \Rightarrow \left(\frac{2}{3}m + m\right)\ddot{x}(t) = mg\sin\alpha(t)$$

$$\ddot{x} = \frac{3}{5}g\,sin\,\alpha(t) \qquad (3)$$

We know that the height change for the plate is equal to the same value in the vertical arm connected to the motor:

$$L\,sin\,\alpha(t) = d\,sin\,\theta\,(t) \qquad (4)$$

$$(2),(3),(4) \Rightarrow \ddot{x} = 0.6g\frac{d}{L}\theta \quad (5)$$

The same is true for the y direction. In general, if we want to express the equations:

$$\begin{cases} \ddot{x} = 0.6g\frac{d}{L}\theta_x \\ \ddot{y} = 0.6g\frac{d}{L}\theta_y \end{cases} \tag{6}$$

2. Electrical Part

The servo motor is used to control the plate in x and y directions.

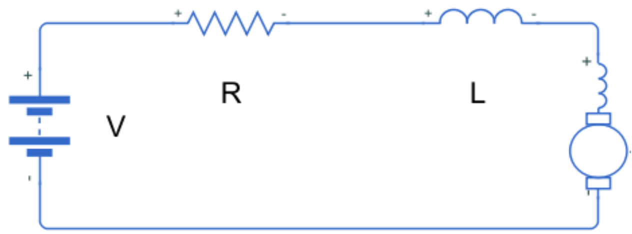


Figure 4. Electrical circuit.

KVL law in the circuit :

$$-V + RI_a + L\frac{dI_a}{dt} + V_b = 0 \Rightarrow \quad V = RI_a + L\frac{dI_a}{dt} + k\frac{d\theta}{dt} \tag{7}$$

$k_t = k_e = k$

For mechanical equation:

$$\sum T = J\ddot{\theta} \tag{8}$$

$$KI_a - C\dot{\theta} = J\ddot{\theta} \tag{9}$$

$$(9),(8) \rightarrow \; J\ddot{\theta} + C\frac{d\theta}{dt} = kI_a \tag{10}$$

Where

J: Moment of inertia of the motor
K: Motor torque constant
I_a: The motor current
C: Frictional torque

From equations (7) and (10), the rest of the equations are also extracted. We generalize for both motors:

corresponding to the motor that controls in the x direction:

$$\begin{cases} V_x = R_x I_{a_x} + L\dfrac{dI_{a_x}}{dt} + k\dfrac{d\theta_x}{dt} \\ J\ddot{\theta}_x + C\dot{\theta}_x = KI_{a_x} \end{cases} \tag{11.1}$$

corresponding to the motor that controls in the Y direction:

$$\begin{cases} V_y = R_y I_{a_y} + L\dfrac{dI_{a_y}}{dt} + k\dfrac{d\theta_y}{dt} \\ J\ddot{\theta}_y + C\dot{\theta}_y = KI_{a_y} \end{cases} \tag{11.2}$$

If we assume that R_x is equal to R_y, we rewrite the equations for both directions:

$$\ddot{x} = 0.6g\frac{d}{L}\theta_x$$
$$\ddot{y} = 0.6g\frac{d}{L}\theta_y$$
$$\dot{I}_x = -\frac{1}{L}\left(k\dot{\theta}_x + RI_x - V_x\right)$$
$$\dot{I}_y = -\frac{1}{L}\left(k\dot{\theta}_y + RI_y - V_y\right)$$
$$\ddot{\theta}_x = \frac{1}{J}(-C\theta_x + KI_x)$$
$$\ddot{\theta}_y = \frac{1}{J}\left(-C\theta_y + KI_y\right)$$

Now we need to define the state variables:

$\theta_x = \boldsymbol{x_1}\,, \dot{\theta}_x = \omega_x = \boldsymbol{x_2}\,, \theta_y = \boldsymbol{x_3}\,, \dot{\theta}_y = \omega_y = \boldsymbol{x_4}\,, I_x = \boldsymbol{x_5}\,, I_y = \boldsymbol{x_6}$
$\boldsymbol{x_7} = v_{x,}\ \boldsymbol{x_8} = x\ ,\ \boldsymbol{x_9} = v_{y,}\ \boldsymbol{x_{10}} = y$
Finally, the equations of state will be as follows:

$$\dot{x}_1 = x_2$$
$$\dot{x}_2 = \frac{1}{J}(-Cx_1 + Kx_5)$$
$$\dot{x}_3 = x_4$$
$$\dot{x}_4 = \frac{1}{J}(-Cx_3 + kx_6)$$
$$\dot{x}_5 = -\frac{1}{L}(Kx_2 + Rx_5 - V_x)$$
$$\dot{x}_6 = -\frac{1}{L}\left(Kx_4 + Rx_6 - V_y\right)$$
$$\ddot{x} = -0.6g\tfrac{d}{L}\theta_x \rightarrow \dot{\boldsymbol{x}}_{\boldsymbol{7}} = -1.96\theta_x = -1.96\boldsymbol{x_1}$$
$$\dot{x}_8 = x_7$$
$$\ddot{x} = -0.6g\tfrac{d}{L}\theta_y \rightarrow \dot{\boldsymbol{x}}_{\boldsymbol{9}} = -1.96\theta_y = -1.96\boldsymbol{x_3}$$
$$\dot{x}_{10} = x_9$$

Considering the following values for the available parameters and having the state equations of the system, the matrices A, B, C, and D are obtained, which are shown as follows using MATLAB software:

C = 0.0035, L = 0.05, K = 0.05; $J = 8 * 10^{-4}\,, R = \ \ 1.2$

$$A = \begin{bmatrix} 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ -4..375 & 0 & 0 & 0 & 62.5 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & -4.375 & 0 & 0 & 62.5 & 0 & 0 & 0 & 0 \\ 0 & -1 & 0 & 0 & -24 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & -1 & 0 & -24 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ -1.96 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & -1.96 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}, B = \begin{bmatrix} 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ -1 & 0 \\ 0 & -1 \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 \end{bmatrix}$$

$$C = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \end{bmatrix}, D = \begin{bmatrix} 0 & 0 \\ 0 & 0 \end{bmatrix}$$

In these equations, u1 is the voltage in the x direction, u2 is the voltage in the y direction. According to the matrix C, it is clear that the outputs are x and y.

Show in MATLAB

```
A =
          x1      x2      x3      x4      x5      x6      x7      x8      x9     x10
 x1        0       1       0       0       0       0       0       0       0       0
 x2   -4.375       0       0       0    62.5       0       0       0       0       0
 x3        0       0       0       1       0       0       0       0       0       0
 x4        0       0  -4.375       0       0    62.5       0       0       0       0
 x5        0      -1       0       0     -24       0       0       0       0       0
 x6        0       0       0      -1       0     -24       0       0       0       0
 x7        0       0       0       0       0       0       0       1       0       0
 x8    -1.96       0       0       0       0       0       0       0       0       0
 x9        0       0       0       0       0       0       0       0       0       1
 x10       0       0   -1.96       0       0       0       0       0       0       0

B =
       u1  u2
 x1     0   0
 x2     0   0
 x3     0   0
 x4     0   0
 x5    -1   0
 x6     0  -1
 x7     0   0
 x8     0   0
 x9     0   0
 x10    0   0
C =
      x1   x2   x3   x4   x5   x6   x7   x8   x9  x10
 y1    0    0    0    0    0    0    1    0    0    0
 y2    0    0    0    0    0    0    0    0    1    0

D =
      u1  u2
 y1    0   0
 y2    0   0
```

First of all, we calculate the eigenvalues of the state matrix to determine the stability state of the system. For this purpose, you can use the command eig (A) in MATLAB.

```
>> eig(A)

ans =

   0.0000 + 0.0000i
   0.0000 + 0.0000i
   0.0000 + 0.0000i
   0.0000 + 0.0000i
 -21.0615+ 0.0000i
  -1.4693 + 1.6813i
  -1.4693 - 1.6813i
 -21.0615+ 0.0000i
  -1.4693 + 1.6813i
  -1.4693 - 1.6813i
```

Well, as it is known, we have four unstable poles located at the origin, which are related to the angle and angular velocity for each motor. We should have expected this because when we give a step input it is uniformly clear that the position and speed go out of range in any direction.

For one thing, we have shown the state variables below so that this phenomenon can be seen intuitively.

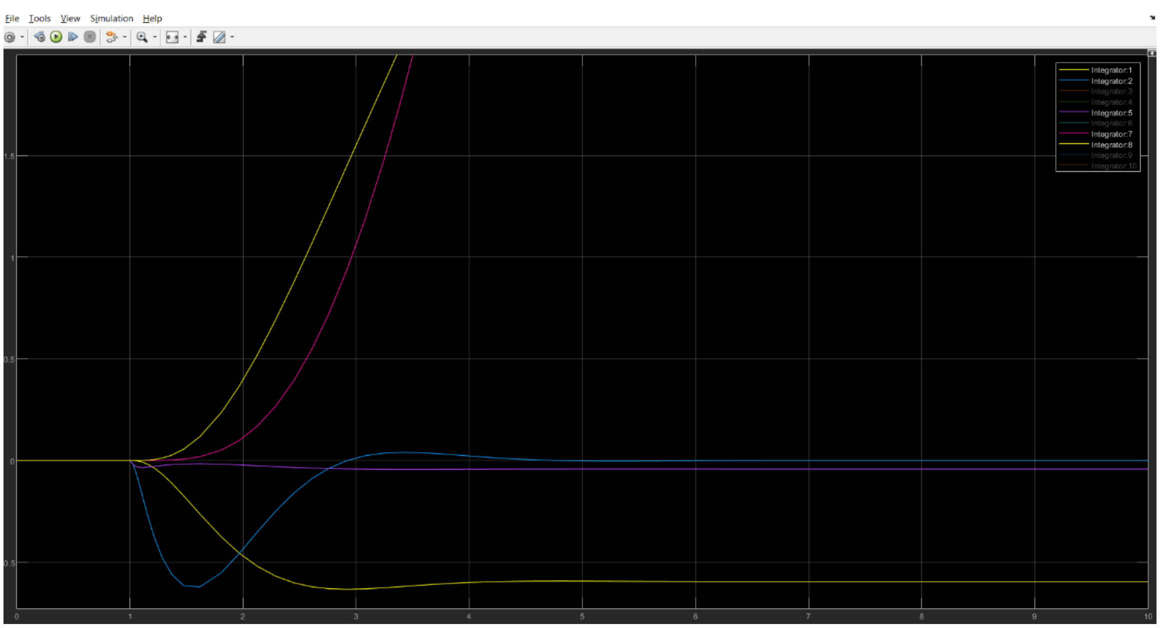


Fig.5. Display the behavior of state variables for step input

It is true that the above system is unstable, but is it possible to control the open loop system?
In the continuation of this project, we try to close the feedback and try to place the unstable dynamic modes of the system in the desired area. But the answer to the question is yes, we can move the ball in a specific direction without closing the feedback loop. The input should be adjusted so that the ball moves according to the equations that are considered for the input functions. Of course, it should be noted that due to the fact that the system is an open loop, it is not at all robust to possible disturbances and will not follow the path.

To demonstrate this, we have tried to adjust the input in such a way that the ball moves on a circular path. We know that for circular movement, it is clear that the position of X and Y (which are also outputs) must change alternately and with a phase difference relative to each other.

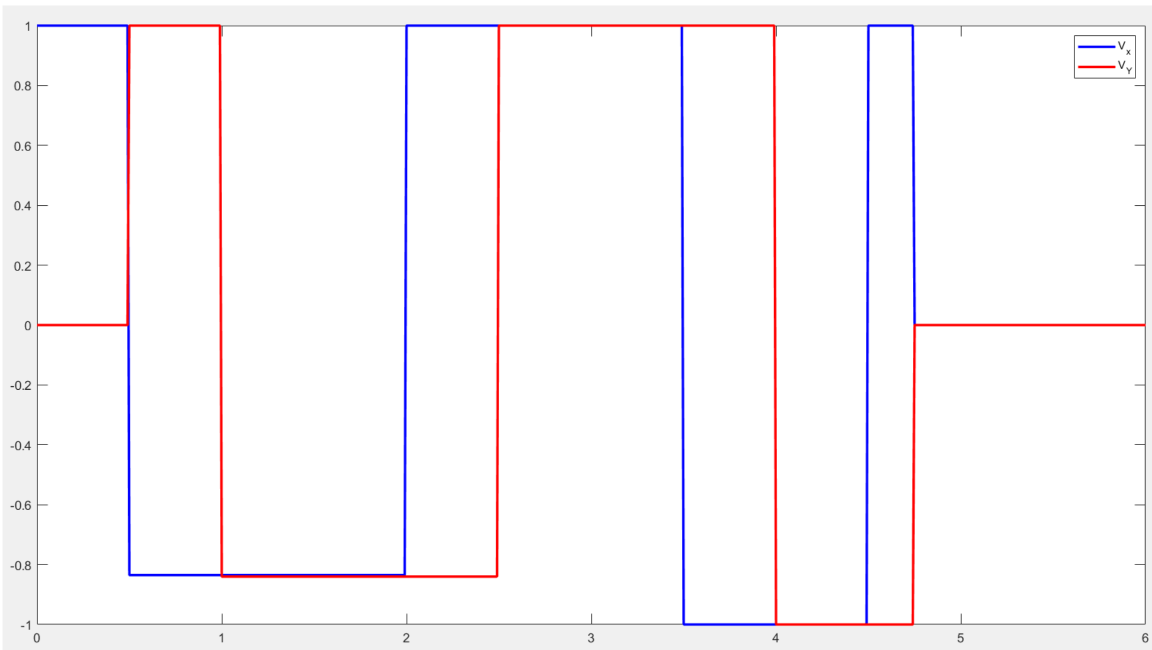


Fig.6. Display specified inputs to create a circular path

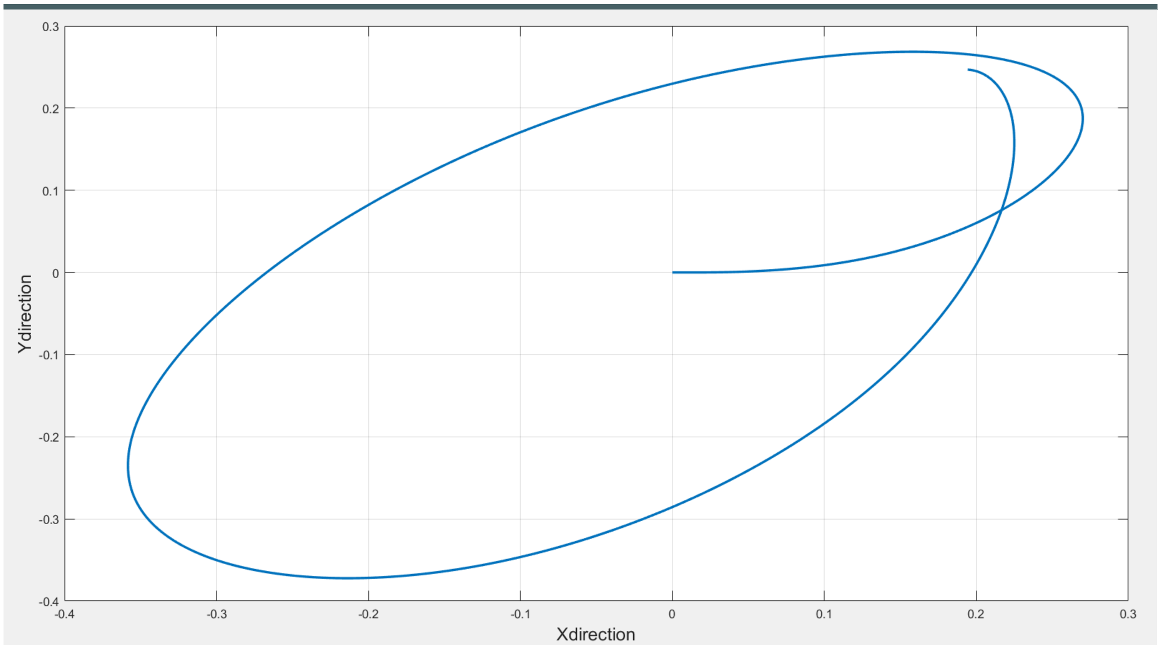


Fig.7. ball path. As you can see, it has started to move from the coordinates (0,0).

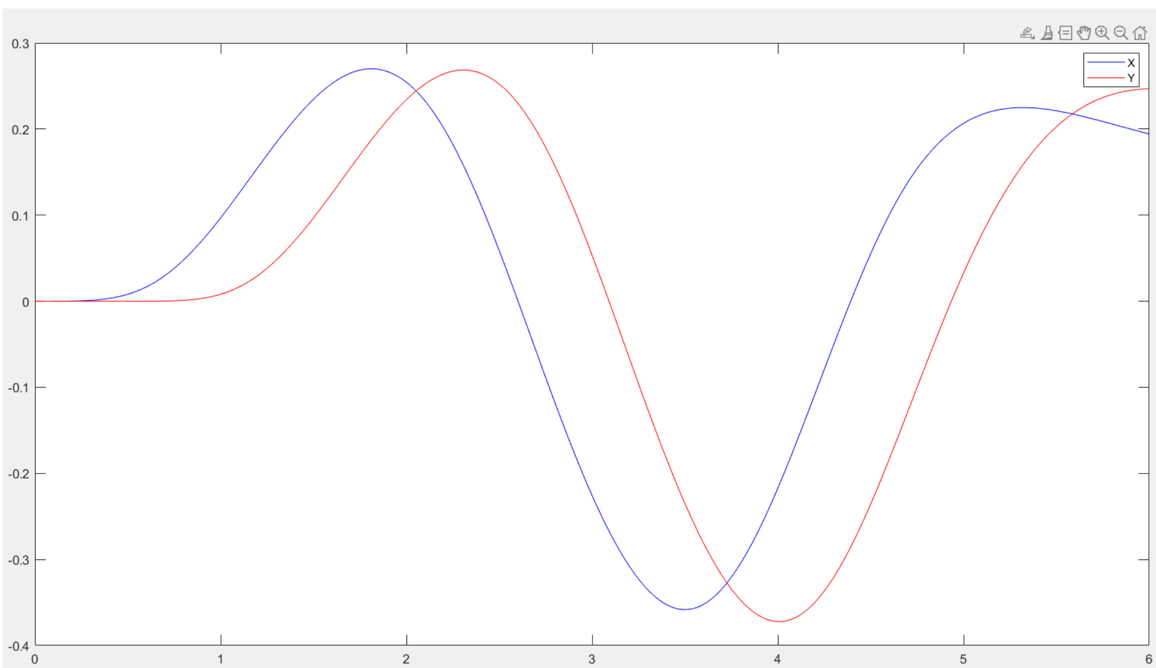


Fig.8. Ball position in any direction. Intermittency and phase difference are evident

Controllability and observability of the system

The first and most important condition for using feedback is the controllability of the system. We also check the observability of the system to know whether the system is observable or not.

1. Controllability

$$\phi_C = [B \;\; AB \;\; A^2B \;\; \cdots \;\; A^{n-1}B]$$

If we check this condition for our system, the obtained matrix is of full rank and therefore the system is controllable.

2. Observability

$$\phi_O = \begin{bmatrix} C \\ CA \\ CA^2 \\ . \\ . \\ CA^{n-1} \end{bmatrix}$$

If we check this condition for our system, the obtained matrix is of full rank and therefore the system is observable.

Realization

Various realizations can be realized for the above system. Among these realizations, we can mention the realizations of controllability, observability, controller, and observer.

For example, we realize controllability realization:

- ✓ First, we obtain the transfer function of the system. We can use " tf " command in MATLAB:

TransferFunction =

From input 1 to output...

1: $\frac{122.5}{s^5+24s^4+66{\cdot}88s^3+105s^2}$

2: 0

From input 2 to output...

1: 0

2: $\frac{122.5}{s^5+24s^4+66{\cdot}88s^3+105s^2}$

According to the existing symmetry, we get the realization for one dimension (or one of the motors that controls one of the directions).

- ✓ The controllable realization will be as follows:

$$\dot{x} = \begin{bmatrix} 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & -105 & -66.88 & -24 \end{bmatrix} x + \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 1 \end{bmatrix} v$$

$$y = [122.5 \quad 0 \quad 0 \quad 0 \quad 0]x$$

This realization is always controllable.

stability of the system

Investigating the stability of the system with different definitions of stability:

1. **BIBO**
   We need poles of the system to check the stability with the criterion of bounded input and bounded output. We have already calculated the poles of the system. We saw that this system has two poles at the origin. Therefore, according to this stability criterion, the open loop system is unstable.

2. **Lyapunov stability**
   To check the stability with this method, we need to check the Lyapunov equation:

$$A^T P + PA = -Q \tag{12}$$

   Where Q is a positive definite matrix. If from the above equation, for the positive definite matrix Q, we can find the matrix P which is positive definite, the system is stable, otherwise, the system will be unstable.
   We can use “lyap” command in MATLAB.
   The P matrix is not positive definite, so the system is unstable:

```
>> Q = eye(5); %Positive definite
>> P = lyap(A',Q)
P =
   1.0e+33 *
    0.0000   -0.0000   -0.0000   -0.0000   -0.0000
   -0.0000   -1.0088   -0.6426   -0.2306   -0.0096
   -0.0000   -0.6426   -0.4093   -0.1469   -0.0061
   -0.0000   -0.2306   -0.1469   -0.0527   -0.0022
   -0.0000   -0.0096   -0.0061   -0.0022   -0.0001
>> eig(P)
ans =
   1.0e+33 *
   -1.4709
   -0.0000
    0.0000
    0.0000
    0.0000
```

State Feedback

This system is not controlled by classical control or in other words by getting feedback from the system output (Figure. 9).

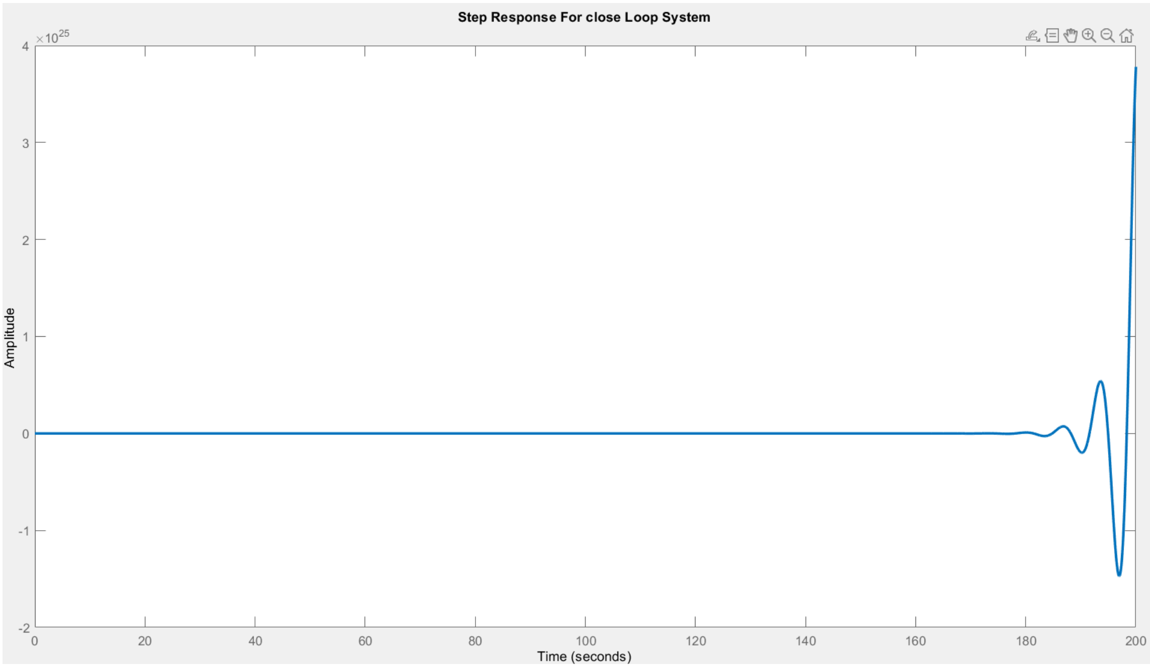


Figure. 9. The system is also unstable with feedback from the output.

Well, using state feedback, we try to place the two unstable poles at the origin in a suitable place. First, we design the feedback mode for the zero reference input (Regulation), then we generalize it for the non-zero input by designing the precompensator (Tracking). The more poles we choose from the origin, the system will be faster we place unstable poles in points $-2 \pm 2i$.

$$\begin{cases} \dot{x} = Ax + Bu \\ y = Cx \end{cases} \tag{13}$$

$$u = -kx \tag{14}$$

Therefore, the closed loop system is as follows:

$$\begin{cases} \dot{x} = (A - Bk)x \\ y = Cx \end{cases} \tag{15}$$

There are various methods to obtain k, methods such as Bass & Gura, direct method, similarity transformation method, Mayne-Murdoch method, and Ackerman formula are among these methods.
In MATLAB, we can use the *place* command to place the poles:

```
K = place(A,B,desired_poles)

K =
  -7.0722  -1.6640  -4.0001   7.7962   6.8574
```

Now we get the closed loop system:

$$A_{CL} = (A - Bk) = \begin{bmatrix} 0 & 1 & 0 & 0 & 0 \\ -4.375 & 0 & 62.5 & 0 & 0 \\ 0 & -1 & -24 & 0 & 0 \\ -1.96 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \end{bmatrix} - \begin{bmatrix} 0 \\ 0 \\ -1 \\ 0 \\ 0 \end{bmatrix} K$$

$$\Rightarrow A_{CL} = \begin{bmatrix} 0 & 1 & 0 & 0 & 0 \\ -4.375 & 0 & 62.5 & 0 & 0 \\ -7.0722 & -2.6640 & -28.0001 & 7.7962 & 6.8574 \\ -1.96 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \end{bmatrix} \qquad (15)$$

The closed loop transfer function will be as follows:

$$G_{CL}(s) = \frac{122.5}{s^5 + 28s^4 + 170.9s^3 + 564.5s^2 + 955s + 840} \quad (16)$$

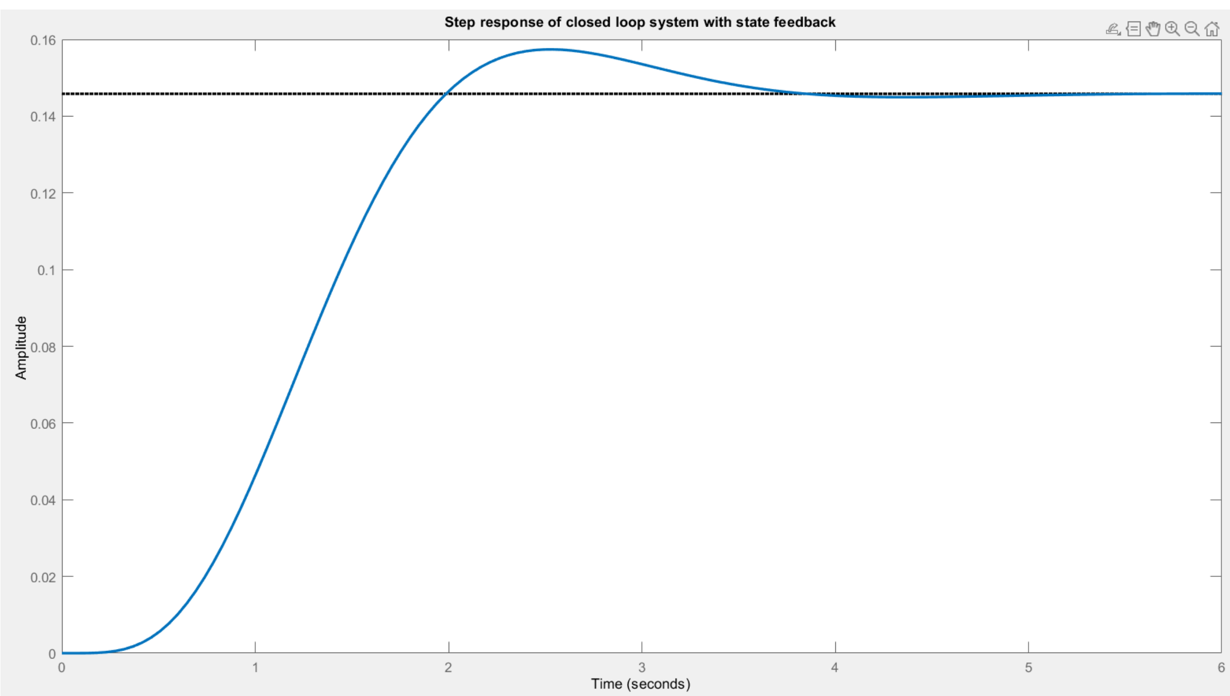


Figure. 10. Step response of closed loop system with state feedback.

As we said before, this design was done to stabilize the (regulation) system. In Figure 10, it is clear that the system is stable, but the steady state has not reached the desired value, which is the same value as the unit. To solve this problem, we need to design a precompensator. In this system, we have chosen static compensator.

$$Pr = G_{cL}(0)^{-1} = [-C(A - Bk)^{-1}B]^{-1}$$

In our system the precompensator value is $\mathrm{Pr} = 6.8574$.

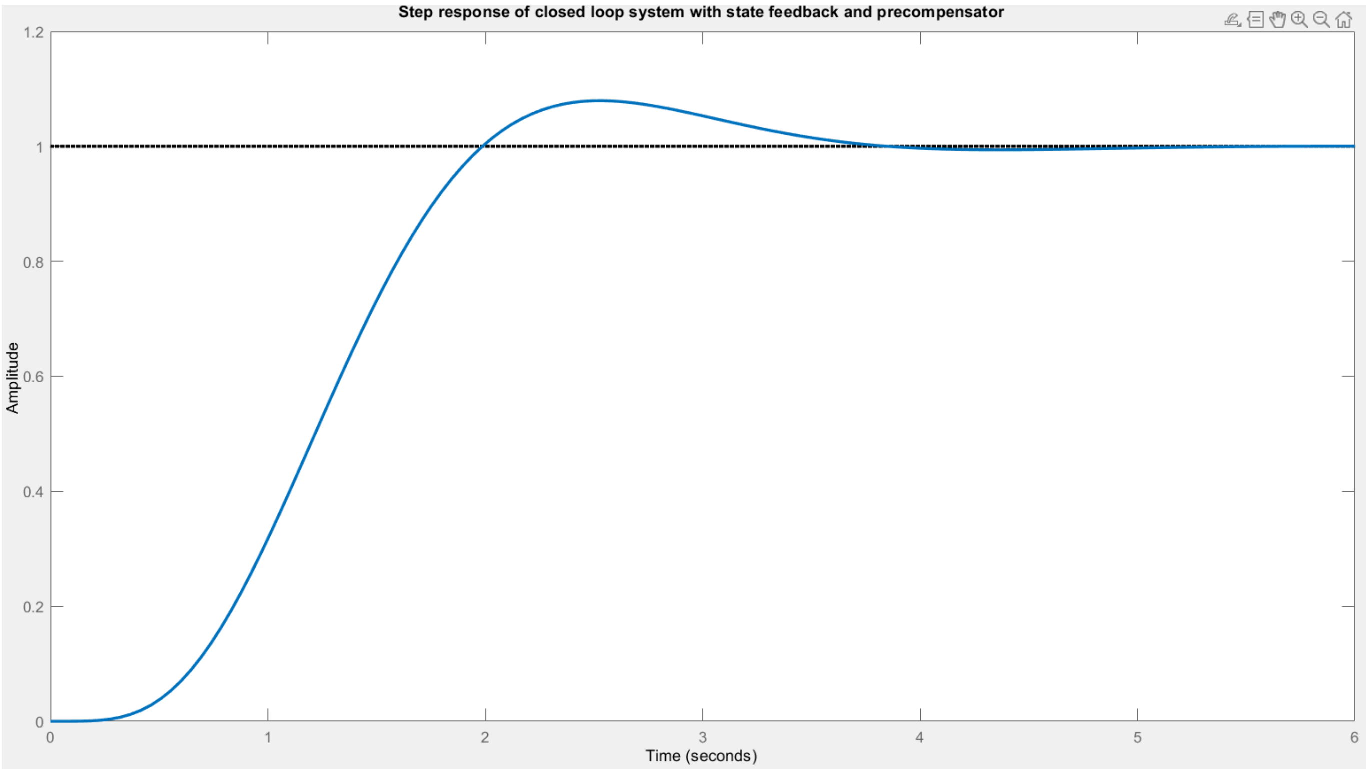


Figure. 10. Step response of closed loop system with state feedback and precompensator.

We were able to stabilize our system. According to effort control, the system can be made faster or slower, the faster the system, the more effort control is needed.
In the following, with the LQR method, we want to establish a suitable balance between control effort and system speed, in other words, we are trying to find an optimal feedback benefit with a suitable stability criterion.

Linear Quadratic Regulation (LQR)

We want to design a state feedback that minimizes the following optimal criterion, in other words, according to the weights of Q and R, it results in the most optimal amount of feedback gain.

$$J = \int_{0}^{\infty} (x^T Q x + u^T R u)\, d\tau \quad (17)$$

where Q and R, are positive semi-definite matrices.

Th state feedback gain is clculated from $k = R^{-1}B^T P$, and to calculate P, we need to solve the Riccati matrix equation, which is as follows:

$$A^T P + PA - PAR^{-1}B^T P + Q = 0 \ (18)$$

For our system we consider R = 1 and Q = eye(5). In MATLAB we can use lqr command to calculate K:

```
>> k = lqr(A,B,Q,R)
k =
   -2.9243   -1.1241   -2.7863    2.4284    1.0000
```

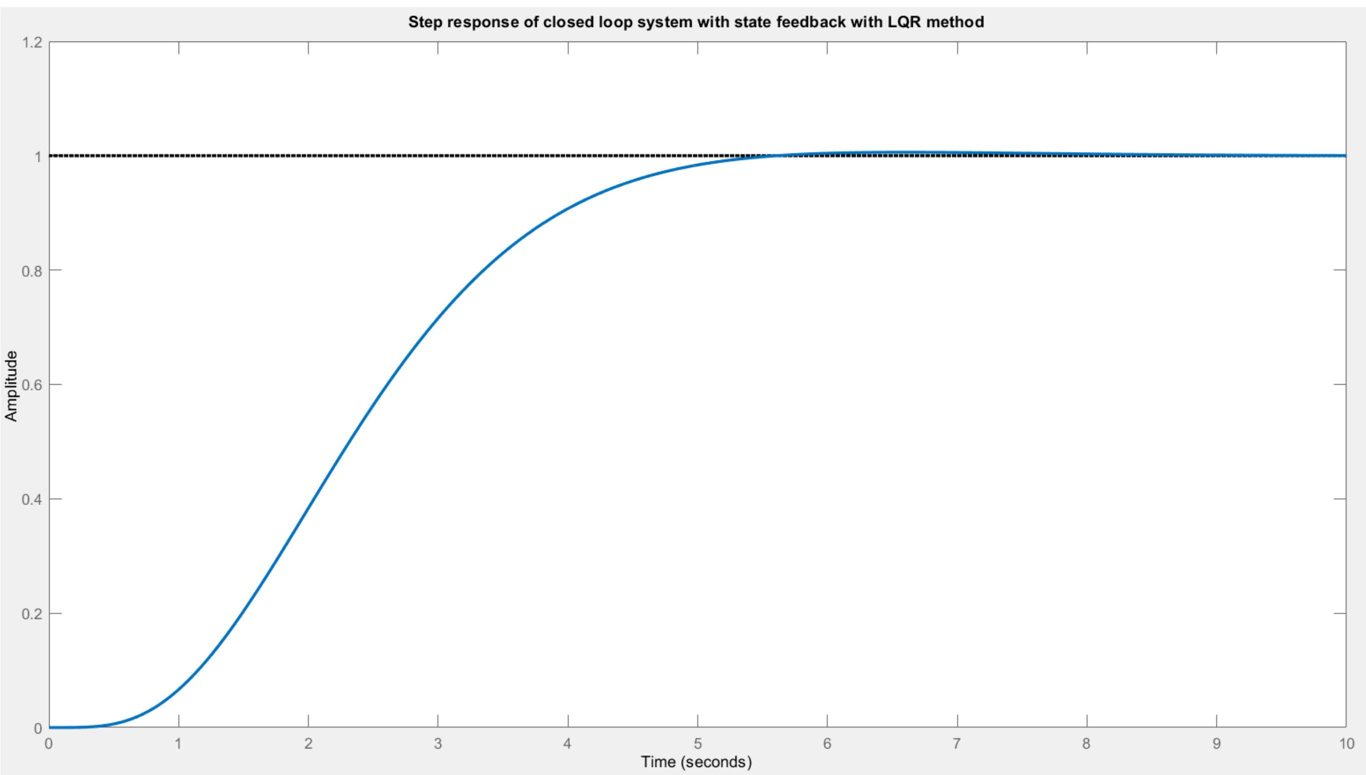


Figure. 11. Step response of closed loop system with state feedback with LQR method.

Conclusion

The ball and plate system control project demonstrates the successful application of modern control theories to a classic control engineering problem. Through meticulous modeling, both mechanical and electrical dynamics were analyzed and combined to form a comprehensive state-space representation of the system. Utilizing state feedback and state-space control methodologies, we were able to stabilize the system and achieve desired performance metrics.

The project employed Linear Quadratic Regulation (LQR) to optimize the balance between system speed and control effort, achieving a robust and efficient control solution. Simulink simulations confirmed the theoretical predictions, illustrating the system's stability and responsiveness under various conditions. This project not only highlights the practical applications of modern control techniques but also reinforces the importance of rigorous mathematical modeling and simulation in the design and analysis of complex control systems.

[1] H. D. Taghirad, *An Introduction to Modern Control*, 3rd ed., K.N. Toosi University of Technology, 2018.
[2] A. Itani, "Ball Plate Balancing System Using Image Processing," M.S. thesis, Dept. Mechatronic Eng., Near East Univ., Nicosia, 2017.
[3] D. Núñez, G. Acosta, and J. Jiménez, "Control of a ball-and-plate system using a state-feedback controller," *Ingeniare. Revista chilena de ingeniería*, vol. 28, no. 1, pp. 6-15, 2020.
[4] R. C. Dorf and R. H. Bishop, "Modern Control Systems," 12th ed., Prentice Hall, 2011.
[5] K. Ogata, "Modern Control Engineering," 5th ed., Prentice Hall, 2010.
[6] B. Friedland, "Control System Design: An Introduction to State-Space Methods," McGraw-Hill, 1986.
[7] MATLAB and Simulink Documentation, The MathWorks, Inc.
[8] Franklin, G.F., Powell, J.D. and Emami-Naeini, A., "Feedback Control of Dynamic Systems," 7th ed., Pearson, 2015.
[9] Bryson, A. E., and Ho, Y. C., "Applied Optimal Control: Optimization, Estimation and Control," Taylor & Francis, 1975.
[10] Slotine, J. J. E., and Li, W., "Applied Nonlinear Control," Prentice Hall, 1991.