Study of the control of a propeller-leviated arm

EFREI Research lab
EFREI - École d'Ingénieurs Généraliste du Numérique

April 23, 2024

Abstract

This report outlines the background, methods, results, and conclusions related to the propeller levitated arm system, utilizing adaptive control techniques such as LQR (Linear Quadratic Regulator) and PID (Proportional-Integral-Derivative). The rationale behind selecting the propeller levitated system was its role as a preliminary study, serving as a comparative analysis for the more intricate quadcopter system due to their similar dynamics.

1 System dynamics, linearization and stability

We want to model the following system :

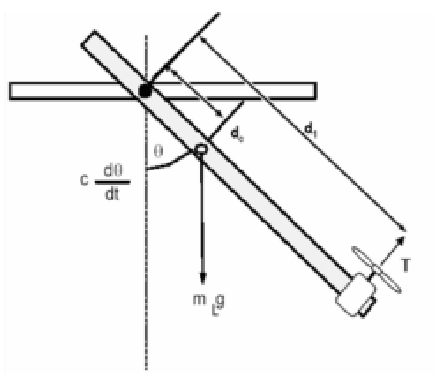

Figure 1: Propeller leviated arm system

Variable	**Value**	**Unit**
c	0.5	Ns/m
m_{rod}	0.8	g
g	9.81	m/s^2
d	1	m
L	2	m
m_{motor}	0.2	g
$thrust$	1	N

Table 1: Variables, Values, and Units

Where c is the damping coefficient, m_{rod} is the mass of the rod, g is the gravitational acceleration, d, is the distance from hinge to the center of gravity of the rod, L is the length of the rod, m_{motor} is the mass of the motor and thrust is the thrust input.

According to Newton's second law of motion :

$$F = ma$$

the torque τ acting on this system is given by

$$\tau = I\ddot{\theta}$$

where I is the moment of inertia and $\ddot{\theta}$ the angular acceleration.
The sum of torques is equal to the net external torque acting on the system, thus we have :

$$I\ddot{\theta} = \tau_{\text{gravity}} + \tau_{\text{thrust}} + \tau_{\text{damping}}$$

.
The moment of inertia for a solid rod rotating about one end is given by

$$I_{\text{rod}} = \frac{1}{3} m_{\text{rod}} L^2$$

.
The moment of inertia for a point mass at the end of a rod is given by

$$I_{\text{motor}} = m_{\text{motor}}.L^2$$

Thus the total moment of inertia is equal to

$$I_{\text{total}} = \frac{1}{3} m_{\text{rod}} L^2 + m_{\text{motor}}.L^2$$

.
The τ_{gravity} is given by the following formula :

$$\tau_{\text{gravity}} = mgdsin(\theta)$$

Where θ is the angle position from the origin.
The τ_{damping} is given by

$$\tau_{\text{damping}} = c\dot{\theta}$$

where $\dot{\theta}$ is the angular velocity.
Thus we obtain the following equation of motion :

$$I\ddot{\theta} + c\dot{\theta} + mgdsin(\theta) = L.Thrust$$

$$\dot{\theta}(0) = 0; \theta(0) = 0$$

The initial angular velocity and angular position are both equal to 0.

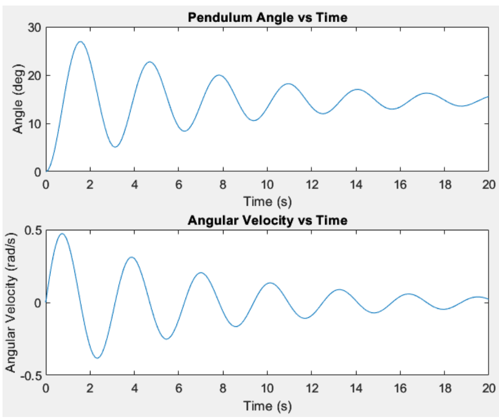

Figure 2: Nonlinear dynamics

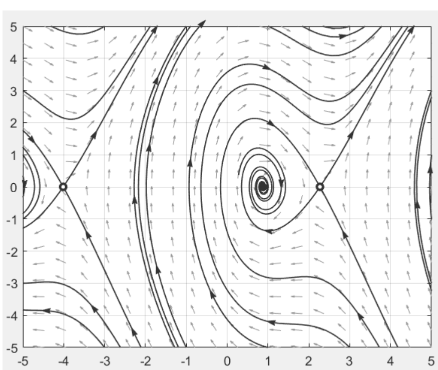

Figure 3: Phase portrait of the nonlinear system

From the phase portrait we deduce from the sink that the system becomes stable as time goes on. It means that the pendulum will slowly reach an equilibrium point after some time. However, if the system receives too much thrust it will blow up to infinity and become unstable.
We are now interested in linearizing the system around a fixed point. Since the control techniques we intend to use later on are methods used for linear systems it is mandatory to linearize the system.

First, we define equation of motion as a system of first order differential equations :

$$\dot{\theta_1} = \theta_2$$

$$\dot{\theta_2} = \frac{mgdsin(\theta_1)}{I} - \frac{C\theta_2}{I}$$

Now, we determine the fixed points :

$$0 = \theta_2$$

$$0 = \frac{mgdsin(\theta_1)}{I} - \frac{C\theta_2}{I} \quad = n\pi$$

From the first equation we find that $\theta_2 = 0$. From the second equation we find that the system has an infinite amount of solutions and thus an infinite number of equilibrium points for any multiple of π. Thus, we can now compute the jacobian matrix of the system :

$$\frac{Df}{D\theta} = \begin{bmatrix} \frac{\partial f_1}{\partial \theta_1} & \frac{\partial f_1}{\partial \theta_2} \\ \frac{\partial f_2}{\partial \theta_1} & \frac{\partial f_2}{\partial \theta_2} \end{bmatrix}$$

With $f_1 = \dot{\theta_1}$ and $f_2 = \dot{\theta_2}$
We get the following jacobian matrix :

$$\frac{Df}{D\theta}(n\pi, 0) = \begin{bmatrix} 0 & 1 \\ -\frac{mgd}{I}cos(n\pi) & -\frac{c}{I} \end{bmatrix}$$

Evaluating at $(n\pi, 0)$ we find that there are two cases to consider : n even and n odd. When n is odd we obtain the following matrix :

$$\frac{Df}{D\theta} = \begin{bmatrix} 0 & 1 \\ -\frac{mgd}{I} & -\frac{c}{I} \end{bmatrix}$$

We can now compute the eigenvalues of this matrix, we find the determinant

$$\Delta = ad - bc$$

we get :

$$\frac{I\lambda^2 + c\lambda + dgm}{I} = 0$$

With the values mentioned in the table above we get the following complex conjugate eigenvalues :

$$\lambda_1 = -0.5357 + 5.7747i \; ; \; \lambda_2 = -0.5357 - 5.7747i$$

The complex part of the eigenvalues display the oscillatory nature of the pendulum. Plus both eigenvalues have a negative real part which means that we have a spiral and that the system is stable.
Now we compute when n is an even number :

$$\frac{Df}{D\theta} = \begin{bmatrix} 0 & 1 \\ \frac{mgd}{I} & -\frac{c}{I} \end{bmatrix}$$

We get the following eigenvalues which are both real with opposite signs

$$\lambda_1 = 5.2885 \ ; \ \lambda_2 = -6.3599$$

Thus we have a saddle and the system is unstable. Which makes sense, when the pendulum is horizontal some force will make it unstable and thus it will exponentially decay until the angle reaches a stable point.

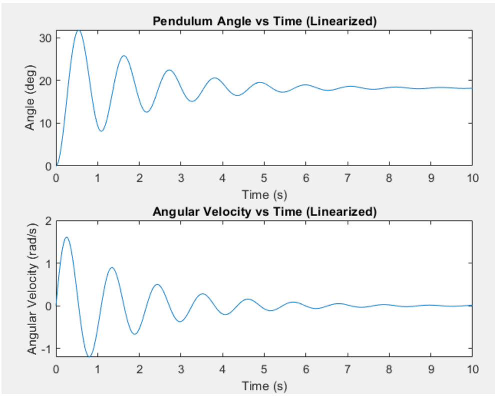

Figure 4: Linearized dynamics

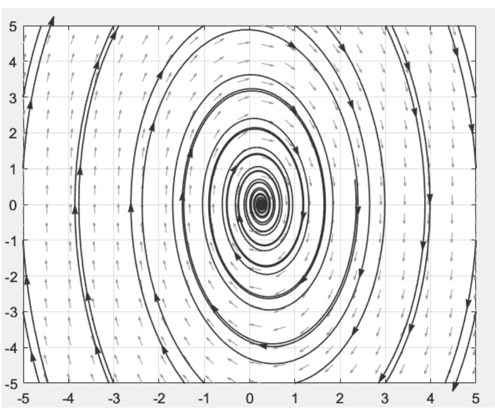

Figure 5: Linearized phase plot

Thus we have the following linearized system :

$$\dot{x} = Ax + Bu$$

$$y = Cx + Du$$

$$\begin{bmatrix} \dot{x_1} \\ \dot{x_2} \end{bmatrix} = \begin{bmatrix} 0 & 1 \\ -\frac{mgd}{I} & -\frac{c}{I} \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} + \begin{bmatrix} 0 \\ \frac{d}{I} \end{bmatrix} u$$

$$\begin{bmatrix} 1 & 0 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} + \begin{bmatrix} 0 \end{bmatrix} u$$

With :
A the state matrix,
B the control matrix,
C the observability matrix,
D the input matrix,
x the state vector,
u the control vector,
y the output vector.

2 Transfer function

Transfer functions are very good tools in control system design as they provide a concise and effective means to represent the relationship between input and output in a system. In order to get the transfer function $H(s)$ we have to apply the Laplace transform of the system's impulse response with the following equation :

$$H(s) = \frac{Y(s)}{U(s)} = C(sI - A)^{-1}B + D$$

. We get the following transfer function :

$$H(s) = \frac{2000}{260s^2 + 200s + 2943}$$

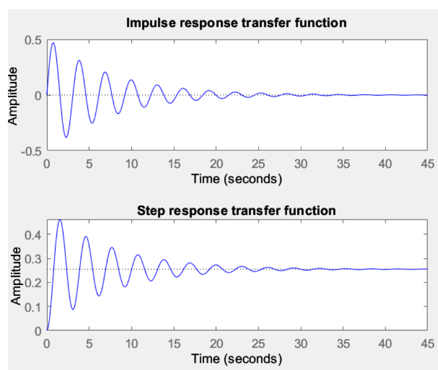

Figure 6: Impulse and step response of H(s)

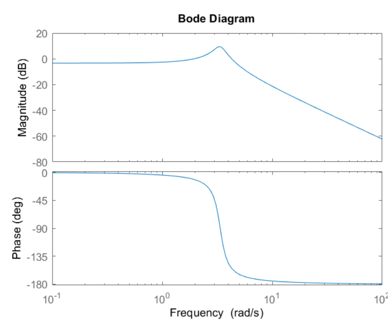

Figure 7: Bode plot of H(s)

As we can see from the bode plot and the impulse response, the system is stable. However, the steady-state response of the system is far from being satisfactory.

3 PID Controller

The PID controller is the most common control law. It is defined by the following equation in the time domain :

$$u(t) = K_p e(t) + K_i \int_0^t e(\tau)\, d\tau + K_d \frac{de(t)}{dt}$$

where u(t) is the control signal at time t,
e(t) is the error signal,
K_p, K_i, K_d are the proportional, integral, and derivative gains respectively.
In the Laplace domaine, the PID controller equation is the following :

$$U(s) = K_p E(s) + K_i \frac{1}{s} E(s) + K_d s E(s)$$

where U(s) is the Laplace transform of the control signal,
E(s) is the Laplace transform of the error signal,
K_p, K_i, K_d are the proportional, integral, and derivative gains respectively

The proportional gain determines how much the controller responds to the current error. The integral term accounts for the cumulative error over time. It integrates the error signal with respect to time, which helps in eliminating steady-state error and improving the system's ability.The derivative term accounts for the rate of change of the error signal. It helps in anticipating future errors by considering the rate at which the error is changing.

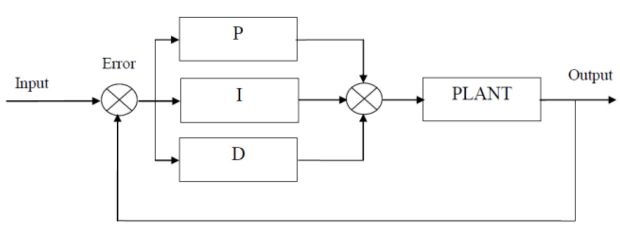

Figure 8: Schematic of PID controller in continous time domain

In the Laplace domain, the system is represented by :

$$T(s) = \frac{H(s)U(s)}{1 + H(s)U(s)}$$

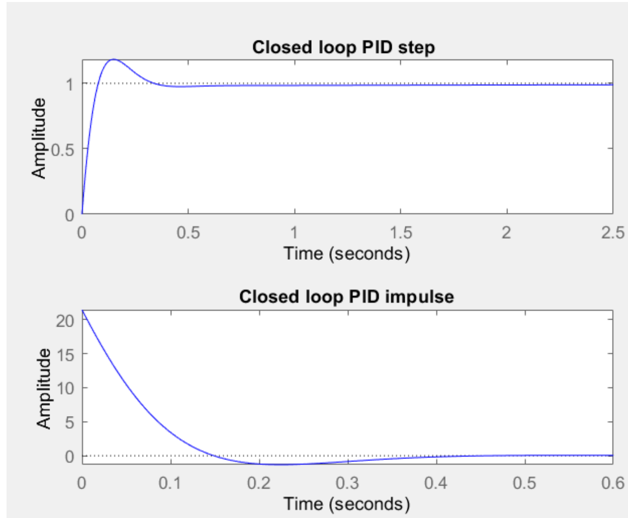

Figure 9: Impulse and step response of T(s)

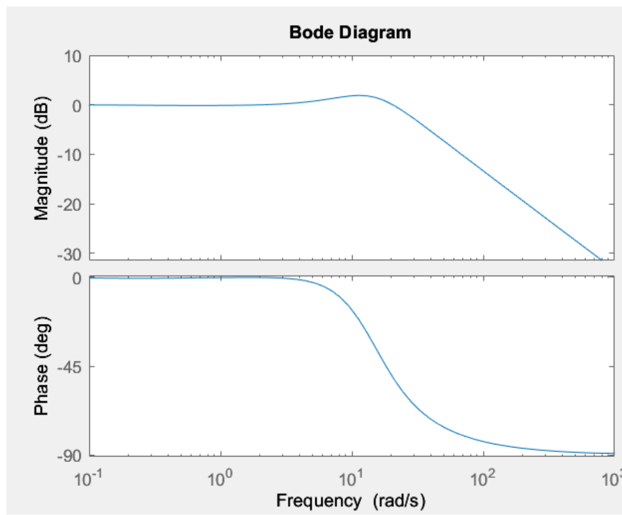

Figure 10: Bode plot of T(s)

As we can see, the response time of the system is now faster, there is a short steady state error which is quickly corrected and the system remains stable.

4 Linear Quadratic Regulator (LQR) State Feedback Design

As seen previously, a system can be expressed as :

$$\dot{x} = Ax + Bu$$

We want to find a state-variable feedback control

$$u = -Kx + v$$

that gives us desirable closed-loop properties. The closed-loop system becomes

$$\dot{x} = (A - BK)x + Bv$$

We want to find the most optimal state-variable feedback, meaning we want to place our poles (eigenvalues) in the most optimal way to reach a desired performance.
Thus, we want to select the state-variable feedback that minimizes the performance of the following equation :

$$J = \frac{1}{2}\int_0^\infty x^T(Q + K^T RK)x, dt$$

The cost function J can be interpreted as an energy function, so that making it small keeps small the total energy of the closed-loop system. It also guarantees that the system will be stable. The two matrices Q and R are chosen to get a specific result. Depending on the parameters the system will exhibit different responses. Generally, larger values of Q will result in the poles of the closed-loop system matrix to be further left in the s-plane. As a result, the system will be much more "aggressive". Larger values of R will result in less control effort which means slower poles.
Q and R have to be respectively positive semi-definite and positive definite. A matrix is positive semi-definite if, for any non-zero column vector the quadratic form $x^T Ax$ is non-negative. In other words, it means that all the eigenvalues of the matrix Q must be non-negative whereas the R matrix needs all its eigenvalues to be positive and non-zeros.
Then we have to solve the algebraic Riccati equation :

$$A^T P + PA - PBR^{-1}B^T P + Q = 0$$

Where is the symmetric positive definite solution matrix, A is the system matrix, B is the input matrix, Q is the state cost matrix and R is the control effort matrix.
Finally, to find K the optimal gain we use the following equation :

$$K = R^{-1}B^T P$$

However, to make sure that we can use the LQR we need to make sure that our system is controllable. For that we use

$$U = [B, AB, A^2B...A^{n-1}B]$$

and we make sure that U the controllability matrix has full rank n.