Model predictive control simulation for a two-link robotic arm

ECE 853 - Optimal control - project report

Sarthak Mehulbhai Desai

Introduction

The development of large-scale production resulting in difficult and repetitive tasks for humans has motivated the increasing interest in arm robotics. The mechanical structure of this class of robots is complex (articulated rigid body) which makes the task of control more difficult. Many approaches are available in the literature to control the robotics arms. Many approaches are available in the literature to control the 2-link manipulator including fuzzy, robust, and adaptive control schemes.

This project deals with a model predictive control (MPC) of a two-link robot arm. The two-link planar manipulator is one of the simplest possible in the class of articulated rigid body robots. The MPC approach presented in this project involves determining model predictive control dynamics of a manipulator robot with two degrees of freedom (DOF). The dynamic model of this robot is nonlinear, so a feedback linearization control is applied to the robot dynamic model to make it linear. Next, based on the obtained linear model, an MPC controller is developed, and a quadratic criterion is minimized.

System model

A plane robot with two degrees of freedom can be presented as depicted in Fig. 1

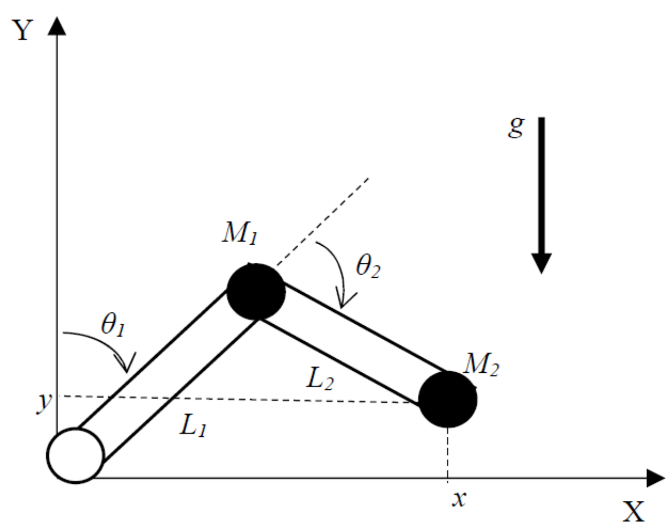


Figure 1

where θ_i, L_i and M_i {i=1,2} are respectively the joint angle, length, and the mass of the first link (i=1) and the second link (i=2). The gravitational force is noted g.

The calculation of the dynamic model of this robot is based on the kinetic and potential energies. These are computed using the direct geometric model (DGM) given by the following formula:

$$\begin{aligned} x &= L_1 \sin(\theta_1) + L_2 \sin(\theta_1 + \theta_2) \\ y &= L_1 \cos(\theta_1) + L_2 \cos(\theta_1 + \theta_2) \end{aligned} \tag{1}$$

Using the equation (1), the total kinetic energy of the two-link robot arm is given by the following equation:

$$\begin{aligned} E = &\frac{1}{2}(M_1 + M_2) L_1^2 \dot{\theta}_1^2 + \frac{1}{2} M_2 L_2^2 \dot{\theta}_1^2 + M_2 L_2^2 \dot{\theta}_1 \dot{\theta}_2 + \\ &\frac{1}{2} M_2 L_2^2 \dot{\theta}_2^2 + M_2 L_1 L_2 \left(\dot{\theta}_1 \dot{\theta}_2 + \dot{\theta}_1^2\right) \cos(\theta_2) \end{aligned} \tag{2}$$

and the potential energy is given by the following formula:

$$\begin{aligned} U = &M_1 g L_1 \cos(\theta_1) + \\ &M_2 g \left(L_1 \cos(\theta_1) + L_2 \cos(\theta_1 + \theta_2)\right) \end{aligned} \tag{3}$$

To find the robot motion equations, we use the formalism of Lagrange:

$$L = E - U \tag{4}$$

The Euler-Lagrange equation can be solved using the Lagrangian L. This equation depends on the partial derivatives of the kinetic and potential energy properties of mechanical systems. By computing these derivatives, we can determine the equations of motion. The definition of the Euler-Lagrange equation is as follows:

$$\tau = \frac{d}{dt}\left(\frac{\partial L}{\partial \dot{\theta}_i}\right) - \frac{\partial L}{\partial \theta_i} \tag{5}$$

where L and $\tau = [\tau_1 \ \tau_2]^T$ are respectively the Lagrangian of the motion and the torques vector.

Developing the equation (5), the dynamic model of a robotic arm with two degrees of freedom (DOF) is given by the following formula:

$$\begin{aligned} &M(\theta)\ddot{\theta} + C(\theta, \dot{\theta}) + G(\theta) = \tau \\ &Y = \theta \end{aligned} \tag{6}$$

Where:

- $\theta = [\theta_1 \quad \theta_2]^{\mathrm{T}}$ is joint variable vector;
- $\tau = [\tau_1 \quad \tau_2]^{\mathrm{T}}$ is torque vector (control input);
- Y is the output vector;
- $G(\theta) = \begin{bmatrix} -(M_1 + M_2) g L_1 \sin(\theta_1) - M_2 g L_2 \sin(\theta_1 + \theta_2) \\ -M_2 g L_2 \sin(\theta_1 + \theta_2) \end{bmatrix}$
  is a vector of gravity torques;
- $C(\theta, \dot{\theta}) = \begin{bmatrix} -M_2 L_1 L_2 \left(2\dot{\theta}_1 \dot{\theta}_2 + \dot{\theta}_1^2\right) \sin(\theta_2) \\ -M_2 L_1 L_2 \dot{\theta}_1 \dot{\theta}_2 \sin(\theta_2) \end{bmatrix}$ represents
  the vector of Coriolis and centrifugal forces;
- $M(\theta) = \begin{bmatrix} D_1 & D_2 \\ D_3 & D_4 \end{bmatrix}$ is the inertia matrix;

with:

$$D_1 = (M_1 + M_2) L_1^2 + M_2 L_2^2 + 2 M_2 L_1 L_2 \cos(\theta_2)$$
$$D_2 = M_2 L_2^2 + M_2 L_1 L_2 \cos(\theta_2)$$
$$D_3 = D_2$$
$$D_4 = M_2 L_2^2$$

Controller Design

This section outlines the development of a predictive control for a two-degree-of-freedom robotic arm. The nonlinear dynamic model described in Equation (6) is considered for this purpose. Our first step is to implement a feedback linearization control to convert the nonlinear model into a linear one. With the linear model in hand, we proceed to design a model predictive control in the second step.

Feedback linearization is a technique used to design a control system for a nonlinear system by transforming it into an equivalent linear system using a feedback controller. One of the key steps in this process is to choose a synthetic control vector, which is a virtual input that allows us to cancel out the nonlinear terms in the system dynamics.

The synthetic control vector is chosen in such a way that its derivative, when multiplied by the system dynamics, cancels out the nonlinear terms in the dynamics.

For developing a feedback linearization control approach of the dynamic model (Equation (6)) of a two-link robot arm, the output Y is differentiated until the control input τ is visible. In this case, τ appears in the second derivative of Y, indicating that the relative degree is two. The formula for the second derivative of Y is as follows:

$$\ddot{Y} = \ddot{\theta} = M(\theta)^{-1}\left(-C(\theta,\dot{\theta}) - G(\theta) + \tau\right) = v \qquad (7)$$

Where $v = [v_1\ v_2]^T$ is a synthetic control vector.

Thus, providing synthetic input v, which is a virtual input that allows us to cancel out the nonlinear terms in the system dynamics.

By rearranging equation (7) we get the following feedback linearization control:

$$\tau = M(\theta)v + C(\theta,\dot{\theta}) + G(\theta) \qquad (8)$$

When we apply the control law presented in equation (8) to the nonlinear system (6) of a manipulator robot with two degrees of freedom, the system becomes a double integrator linear system. The relative degree of this linear system is two. Consequently, the control law (8) achieves complete linearization of the nonlinear system (6) and results in a linear system for each joint variable.

$$\frac{\theta_1(s)}{v_1(s)} = \frac{1}{s^2} \text{ and } \frac{\theta_2(s)}{v_2(s)} = \frac{1}{s^2} \qquad (9)$$

where s is a Laplace variable, θ is the output of the linearized system and v is the input.

The linearization of the nonlinear system has been done. So, we can develop minimum time control for the two-link robot arm.

Model Predictive Control

In the case of a robot arm with two DOF and after application of the feedback linearization (8) to the nonlinear system (6), we obtain the following two decoupled linear systems.

$$\begin{aligned} \ddot{\theta}_1 &= v_1 \\ \ddot{\theta}_2 &= v_2 \end{aligned} \qquad (10)$$

Considering the first equation of (10). This system can be rewritten in the state-space form:

$$\begin{aligned} \dot{x}_1(t) &= x_2(t) \\ \dot{x}_2(t) &= v_1(t) \\ Y(t) &= x_1(t) \end{aligned} \qquad (11)$$

Where $[x_1\ x_2]^T = [\tau_1\ \tau_2]^T$ and v_1 is a synthetic control of the first link of the robot and Y is the output.

Now, we will develop a model predictive controller (MPC) for the first link of the robot arm. The same process will be applied to develop an MPC controller for the second link of the robot. Assuming a constant value for the variable $v_1(t) = v_1$ over the time interval [t, t+h] (where h represents the prediction horizon time), and utilizing equation (11), we can formulate the prediction model as follows:

$$\dot{\theta}_1(t+h) = v_1 h + \dot{\theta}_1(t)$$
$$\theta_1(t+h) = \frac{1}{2} v_1 h^2 + \dot{\theta}_1(t) h + \theta_1(t) \tag{12}$$

Now, given the reference angle of the first link θ_{1d} (constant), the proposed one-horizon time quadratic cost function for stabilizing the system is defined by:

$$J = e_1^2(t+h) + \rho \dot{e}_1^2(t+h) \tag{13}$$

Where $e_1(t+h) = \theta_{1d} - \theta_1(t+h)$ is the predicted angle error, $\dot{e}_1(t+h) = 0 - \dot{\theta}_1(t+h)$ is the predicted velocity error. The horizon time h and the weight factor ρ are both positive parameters to be determined later.

Substituting the prediction model (12) into (13) and minimizing the criterion J with respect to v_1 the obtained MPC controller, is given by the following formula:

$$v_1(t) = k_3\theta_{1d} - k_1\theta_1(t) - k_2\dot{\theta}_1(t) \tag{14}$$

With the following control gains:

$$k_1 = k_3 = \frac{2}{h^2 + 4\rho} \text{ and } k_2 = \frac{2h^2 + 4\rho}{h^3 + 4\rho h} \tag{15}$$

The block diagram of the closed-loop system can be presented as depicted in Fig. 2

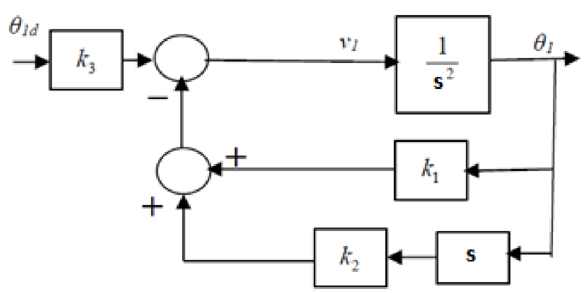


Figure 2

Note that the MPC controller is obtained by minimizing the cost function with respect to the synthetic input v. Here the synthetic input is equal to $\ddot{\theta}$ which is the angular acceleration of each link around its corresponding joint. Hence, we are minimizing the error in angular position and velocity with respect to the angular acceleration. As rate of change of angular velocity and the rate of change of angular position are proportional to the angular acceleration. To minimize the cost function the proposed MPC controller tries to limit the angular acceleration.

The angular acceleration of the links (around their corresponding joints) are directly controlled by the input – Torques τ (by τ = I $\ddot{\theta}$. I = moment of inertia of the link). As the angular acceleration is constrained, this results in the control input (Torques) to also be constrained by the controller. Thus, the even without explicit constraints on the input torques, the way the MPC controller is formulated, ensures that the control input do not saturate to their maximum value (mathematically infinity)

Concerning the response of the system, we would like to have a behavior similar to a system of the second order:

$$\frac{{w_0}^2}{s^2 + 2\zeta w_0 s + {w_0}^2} \tag{16}$$

Where ζ is the damping factor and w_0 is the natural frequency. The transfer function of the system presented in Fig. 2 is given by:

$$\frac{\theta_1(s)}{\theta_{1d}(s)} = \frac{K_3}{s^2 + K_2 s + K_1} \tag{17}$$

From (15), (16) and (17) we derive:

$$2\zeta w_0 = \frac{2h^2 + 4\rho}{h^3 + 4\rho h} \tag{18}$$

$$w_0^2 = \frac{2}{h^2 + 4\rho} \tag{19}$$

The weight factor thus is given as:

$$\rho = \frac{2 - (w_0 h)^2}{4 w_0^2} \tag{20}$$

And the horizon time h can be found using:

$$w_0^2 h^2 - 4\zeta w_0 h + 2 = 0 \tag{21}$$

h and ρ are both real and positive.

Simulation validation

Choosing $\zeta = \sqrt{2} / 2$ and $W_0 = 4$ rads/sec, we determine the parameter h from (21). Now, the horizon time obtained is used in Eq. (20) to determine the weight factor ρ.

For simulation purpose, we assume that the mass and the length of the first and the second links of the robot arm are $M_{i=(1,2)} = 1$(kg) and $L_{i=(1,2)} = 1$ (m), respectively. The initial and the desired orientations of the first and the second links of the robot arm are chosen to be:

(i) $\theta_1(0) = \pi/2$, $\theta_2(0) = -\pi/2$, and $\theta_{1d} = \pi/4$ and $\theta_{2d} = -\pi/4$
(ii) $\theta_1(0) = \pi/2$, $\theta_2(0) = -\pi/2$, and $\theta_{1d} = -\pi/2$ and $\theta_{2d} = \pi/2$

Respectively for two cases.

According to the obtained horizon time and the weight factor, the gains of the MPC controller are:

$K_1 = 15.99$, $K_2 = 5.65$ and $K_3 = K_1$.

The Simulink model designed with the proposed MPC controller is illustrated in figure 3.

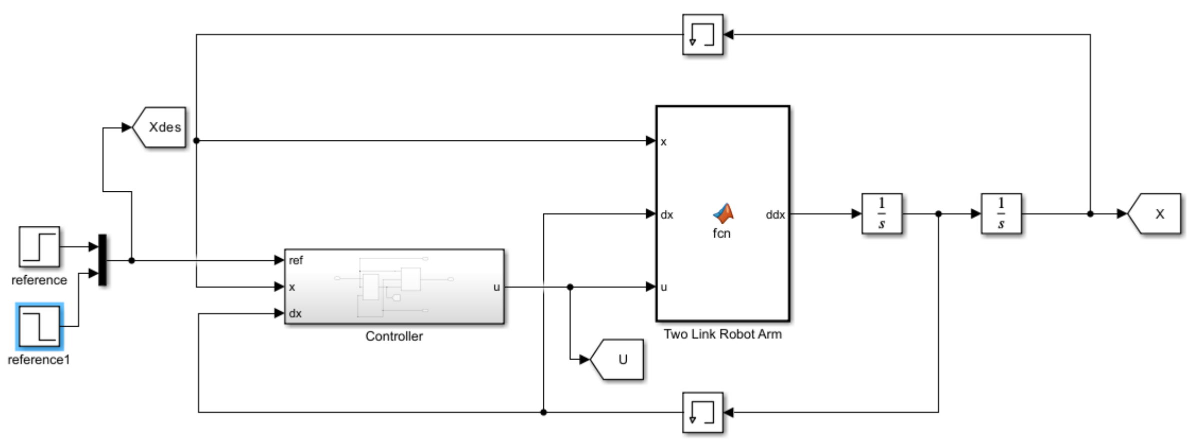


Figure 3

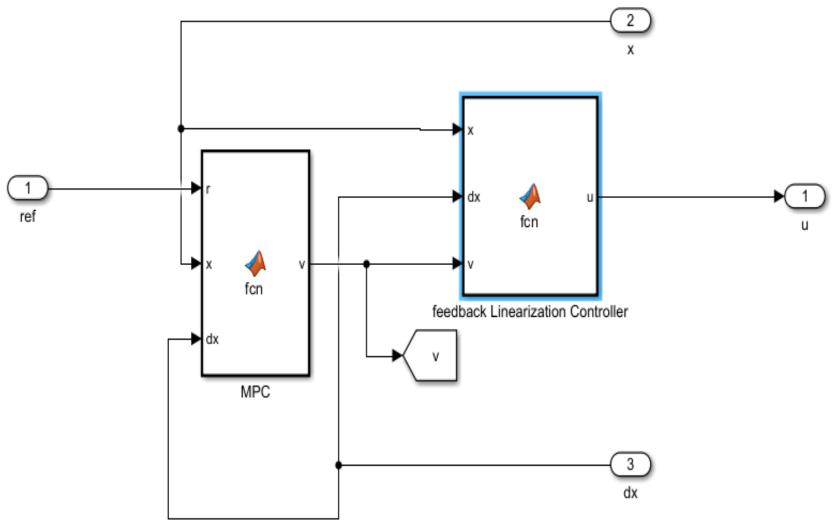
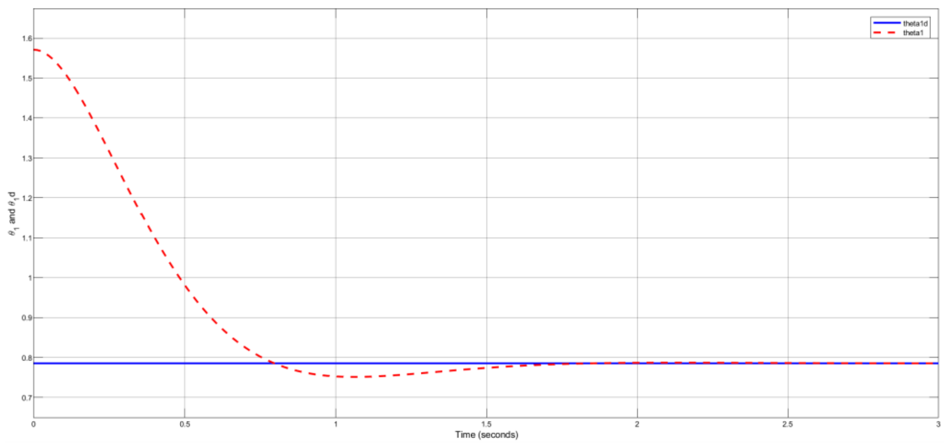


Figure 4

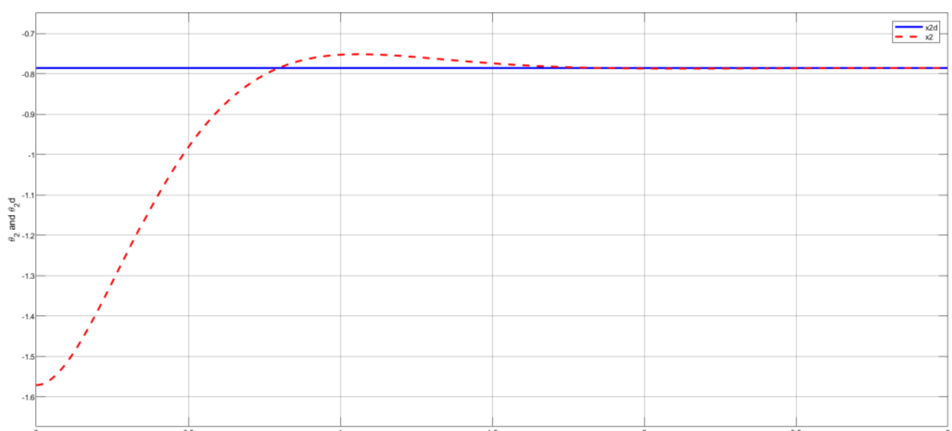
The script used for the Robot model, feedback linearization controller and MPC is provided in appendix A.

(i) $\theta_1(0) = \pi/2$, $\theta_2(0) = -\pi/2$, $\theta_{1d} = \pi/4$ and $\theta_{2d} = -\pi/4$

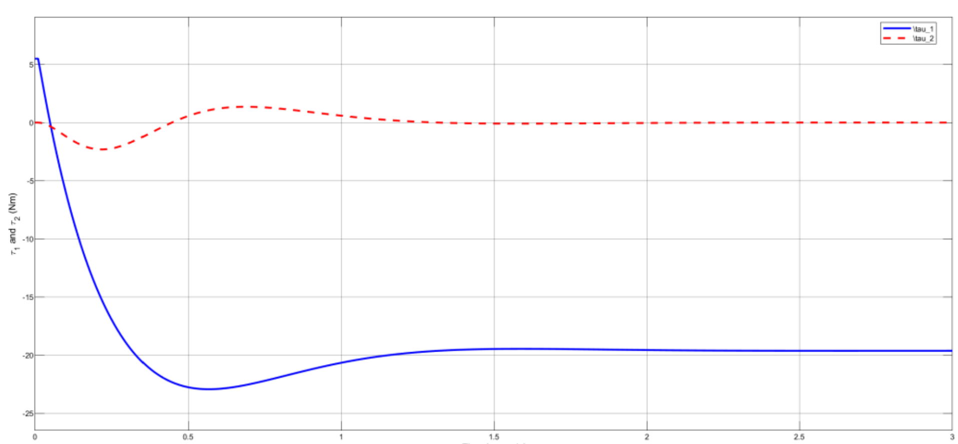
Θ1 and Θ1d:



Θ2 and Θ2d:

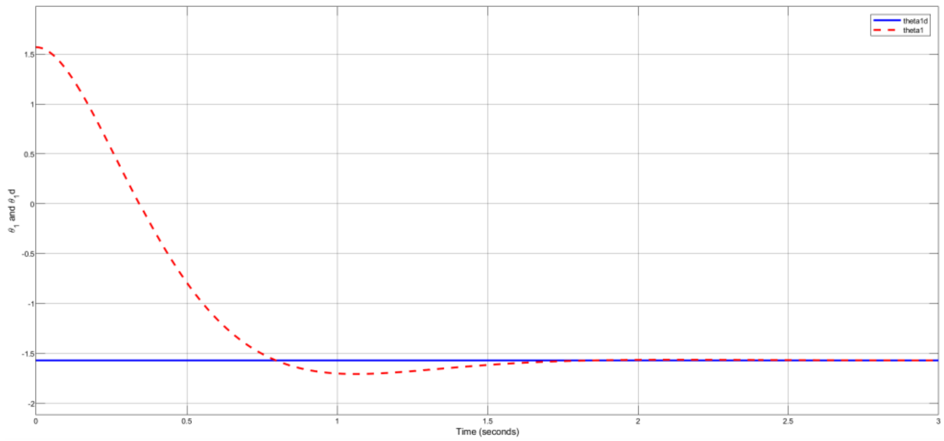


τ1 and τ2:

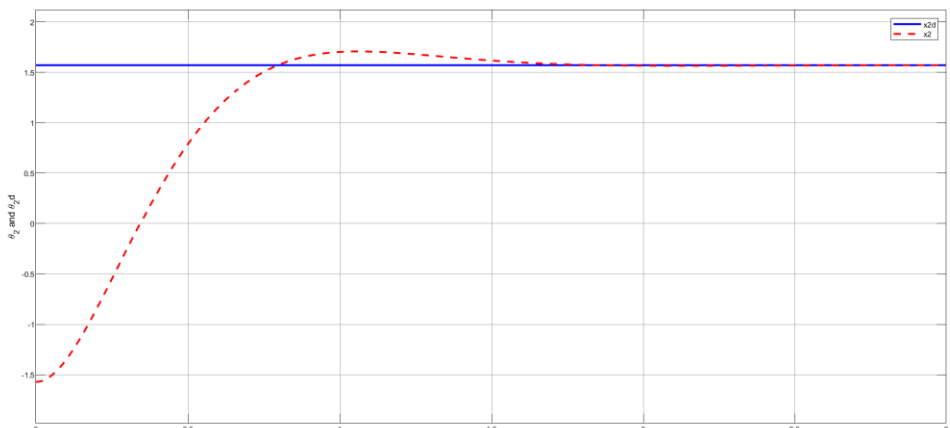


(ii) $\theta_2(0) = \pi/2$, $\theta_2(0) = -\pi/2$, $\theta_{1d} = -\pi/2$ and $\theta_{2d} = \pi/2$

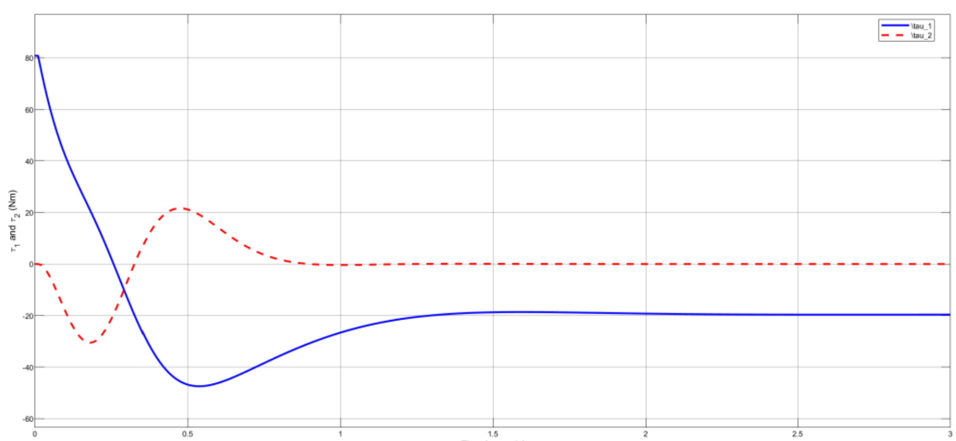
Θ1 and Θ1d:



Θ2 and Θ2d:

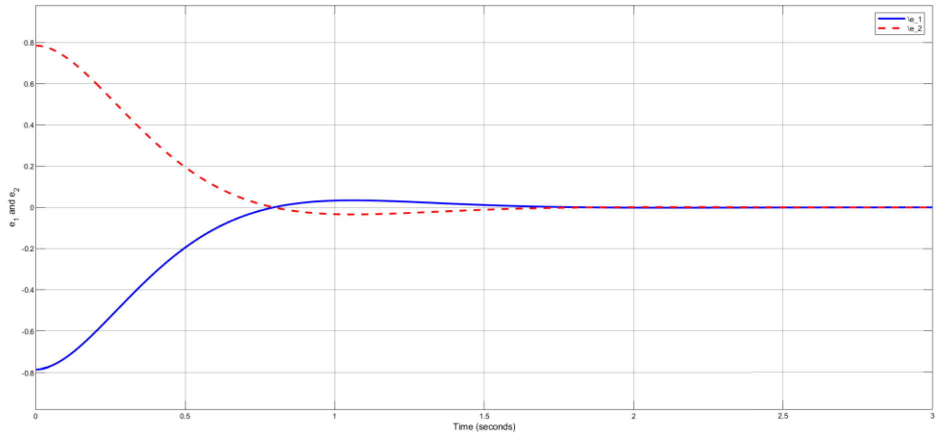


τ1 and τ2:

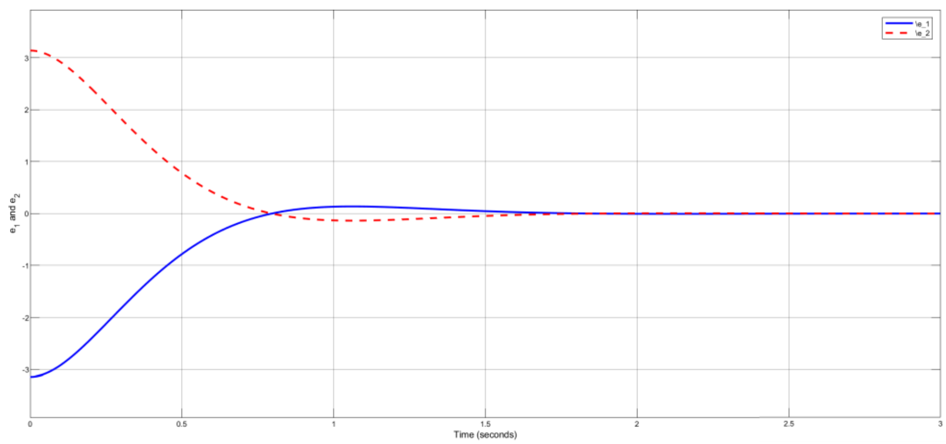


The error for both cases minimizes to 0 in about the same time of 2 seconds.

(i) e_1 and e_2 for $\theta_1(0) = \pi/2$, $\theta_2(0) = -\pi/2$, $\theta_{1d} = \pi/4$ and $\theta_{2d} = -\pi/4$



(i) e_1 and e_2 for $\theta_1(0) = \pi/2$, $\theta_2(0) = -\pi/2$, $\theta_{1d} = -\pi/2$ and $\theta_{2d} = \pi/2$



We can notice fast and asymptotic convergence of both joint variables from the prior figures. The synthetic controls reach zero when the end effector of the robot reaches its objective.

The figures show the robot torques (τ_1 and τ_2) that can be obtained from synthetic controls, The convergence of the joint angle errors (e_1 and e_2) of the two-link of the robot arm towards zero using the proposed approach of control.

conclusions.

In this project, a model predictive control (MPC) strategy is introduced for a two-degree-of-freedom robotic arm. The method involves linearizing the robot's nonlinear dynamic model using feedback linearization control, followed by developing an MPC controller based on the resulting linear model. By calculating the time horizon and weight factor, the closed loop system behaves like a second-order system. The simulation results demonstrate the effectiveness of the proposed approach. To enhance system performance, future work can focus on incorporating the control horizon into the cost function. The validation of the approach on a physical robot can also be planned.

References.

- J. Wilson, M. Charest and R. Dubay, “Non-linear model predictive control schemes with application on a 2 link vertical robot manipulator”, Robotics and Computer-Integrated Manufacturing, vol. 41, pp. 23–30, October 2016.
- Efficient Model Predictive ControlAlgorithm for AircraftPeter W. Gibbens∗ and Eran D. B. Medagoda University of Sydney,Sydney, New South Wales 2006, Australia (PDF) Efficient Model Predictive Control Algorithm for Aircraft. Available from: https://www.researchgate.net/publication/270872533_Efficient_Model_Predictive_Control_Algorithm_for_Aircraft [accessed Apr 26 2023].
- A. Mills, A. Wills, B. Ninness, “Nonlinear model predictive control of an inverted pendulum”, Proceedings of the American Control Conference, St. Louis, pp. 2335�2340, 2009.
- E. -H. Guechi, S. Bouzoualegh, L. Messikh and S. Blažic, "Model predictive control of a two-link robot arm," 2018 International Conference on Advanced Systems and Electric Technologies (IC_ASET), Hammamet, Tunisia, 2018, pp. 409-414, doi: 10.1109/ASET.2018.8379891.
- M. Zheng, W. Liao, C. Yin and A. Wang, "Nonlinear Tracking Control Design of A Robot Arm Using Robust Right Coprime Factorization and Sliding Mode Approaches", Proceedings of the 2014 International Conference on Advanced Mechatronic Systems, pp. 11-16, August 2014, Japan.

APPENDIX A

2-Link arm dynamics block MATLAB script:

```
function ddx = fcn(x,dx,u)

M1 = 1;
M2 = 1;
L1 = 1;
L2 = 1;
g = 9.81;
G = [-(M1+M2)*g*L1-M2*g*L2*sin(x(1)+x(2));...
    -M2*g*L2*sin(x(1)+x(2));];
C = [M1*L1*L2*(2*dx(1)*dx(2) + dx(1)^2)*sin(x(2));...
    -M2*L1*L2*dx(1)*dx(2)*sin(x(2));];
M = [-(M1+M2)*L1^2 + M2*L2^2 + 2*M2*L1*L2*cos(x(2)), M2*L2^2 + M2*L1*L2*cos(x(2));...
     M2*L2^2 + M2*L1*L2*cos(x(2)),  M2*L2^2;];
ddx = inv(M)*(-C-G+u);

```

Feedback linearization control block:

```
function u = fcn(x,dx,v)

M1 = 1;
M2 = 1;
L1 = 1;
L2 = 1;
g = 9.81;
G = [-(M1+M2)*g*L1-M2*g*L2*sin(x(1)+x(2));...
    -M2*g*L2*sin(x(1)+x(2));];
C = [M1*L1*L2*(2*dx(1)*dx(2) + dx(1)^2)*sin(x(2));...
    -M2*L1*L2*dx(1)*dx(2)*sin(x(2));];
M = [-(M1+M2)*L1^2 + M2*L2^2 + 2*M2*L1*L2*cos(x(2)), M2*L2^2 + M2*L1*L2*cos(x(2));...
     M2*L2^2 + M2*L1*L2*cos(x(2)),  M2*L2^2;];



u = M*v + C + G;

```

MPC block:

```
function v  = fcn(r,x,dx)

k1 = 15.99;
k2 = 5.65;
k3 = k1;

v = [k3*r(1) - k1*x(1) - k2*dx(1);...
     (k3*r(2) - k1*x(2) - k2*dx(2));];

```